

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of

Fumio NAGASAKA

Application No.: 09/619,607

Filed: July 19, 2000

Docket No.: 106365

For: LOGGED-IN DEVICE AND LOG-IN DEVICE

CLAIM FOR PRIORITY

Director of the U.S. Patent and Trademark Office
Washington, D.C. 20231

Sir:

The benefit of the filing date of the following prior foreign application filed in the following foreign country is hereby requested for the above-identified patent application and the priority provided in 35 U.S.C. §119 is hereby claimed:

Japanese Patent Application No. 11-219858 filed August 3, 1999.

In support of this claim, a certified copy of said original foreign application:

 X is filed herewith.

 was filed on in Parent Application No. filed .

It is requested that the file of this application be marked to indicate that the requirements of 35 U.S.C. §119 have been fulfilled and that the Patent and Trademark Office kindly acknowledge receipt of this document.

Respectfully submitted,



James A. Oliff
Registration No. 27,075

Thomas J. Pardini
Registration No. 30,411

JAO:TJP/emb

Date: September 19, 2000

OLIFF & BERRIDGE, PLC
P.O. Box 19928
Alexandria, Virginia 22320
Telephone: (703) 836-6400

| |
|--|
| DEPOSIT ACCOUNT USE AUTHORIZATION Please grant any extension necessary for entry; Charge any fee due to our Deposit Account No. 15-0461 |
|--|

日 本 国 特 許 庁

PATENT OFFICE
JAPANESE GOVERNMENT



別紙添付の書類に記載されている事項は下記の出願書類に記載されて
る事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed
in this Office.

出 願 年 月 日

Date of Application:

1999年 8月 3日

出 願 番 号

Application Number:

平成11年特許願第219858号

出 願 人

Applicant(s):

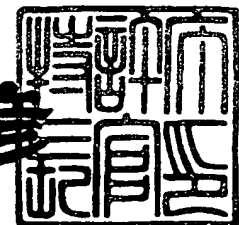
セイコーエプソン株式会社

CERTIFIED COPY OF
PRIORITY DOCUMENT.

2000年 6月29日

特許庁長官
Commissioner,
Patent Office

近 藤 隆 彦



出証番号 出証特2000-3050410

【書類名】 特許願

【整理番号】 PA04C931

【提出日】 平成11年 8月 3日

【あて先】 特許庁長官 伊佐山 建志 殿

【国際特許分類】 G06F 13/00

【発明者】

 【住所又は居所】 長野県諏訪市大和三丁目3番5号 セイコーエプソン株式会社内

 【氏名】 長坂 文夫

【特許出願人】

 【識別番号】 000002369

 【氏名又は名称】 セイコーエプソン株式会社

【代理人】

 【識別番号】 100096817

 【弁理士】

 【氏名又は名称】 五十嵐 孝雄

 【電話番号】 052-218-5061

【選任した代理人】

 【識別番号】 100097146

 【弁理士】

 【氏名又は名称】 下出 隆史

【選任した代理人】

 【識別番号】 100102750

 【弁理士】

 【氏名又は名称】 市川 浩

【選任した代理人】

 【識別番号】 100109759

 【弁理士】

 【氏名又は名称】 加藤 光宏

【手数料の表示】

【予納台帳番号】 007847

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9502061

【包括委任状番号】 9904030

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 被ログイン装置、ログイン装置、及びそれらを備える装置間通信システム、ログイン制御方法、並びに記録媒体

【特許請求の範囲】

【請求項 1】 複数の特定装置から所定の通信路を介してログインされ得る被ログイン装置であって、

或る特定装置からログイン要求を受けた場合に、既にログインしている前記特定装置の数が予め定められたログイン許容数に達している際には、ログイン要求を受けた前記特定装置に対して、ログイン不成立の応答をする応答手段と、

ログイン不成立の応答をする場合に、ログイン要求を受けた前記特定装置に対して、該特定装置が再度ログイン要求を出す場合の再要求タイミングを指示する再要求タイミング指示手段と、

を備えることを特徴とする被ログイン装置。

【請求項 2】 請求項 1 に記載の被ログイン装置において、前記再要求タイミング指示手段は、

ログイン要求を受けた前記特定装置について、その優先順位を設定する優先順位設定手段と、

設定された前記優先順位に従って、前記特定装置についての前記再要求タイミングを決定する再要求タイミング決定手段と、

を備えることを特徴とする被ログイン装置。

【請求項 3】 請求項 2 に記載の被ログイン装置において、前記優先順位設定手段は、

最初のログイン要求を受けた順番に、前記特定装置の優先順位を設定することを特徴とする被ログイン装置。

【請求項 4】 請求項 2 または請求項 3 に記載の被ログイン装置において、前記再要求タイミング決定手段は、設定された前記優先順位の高い特定装置ほど、前記再要求タイミングを短くするように決定することを特徴とする被ログイン装置。

【請求項 5】 請求項 1 に記載の被ログイン装置において、

前記再要求タイミング指示手段は、前記再要求タイミングを、再度ログイン要求を出すまでの時間間隔を表す時定数として指示することを特徴とする被ログイン装置。

【請求項 6】 請求項 1 ないし請求項 5 のうちの任意の 1 つに記載の被ログイン装置において、

前記特定装置から互いに独立にログインされ得る 1 つ以上のロジカル・ユニットを備え、

前記応答手段は、或る特定装置から或るロジカル・ユニットにログイン要求を受けた場合に、既にそのロジカル・ユニットにログインしている前記特定装置の数がそのロジカル・ユニットについて予め定められたログイン許容数に達している際には、ログイン要求を受けた前記特定装置に対して、そのロジカル・ユニットについてのログイン不成立の応答をすると共に、

前記再要求タイミング指示手段は、そのロジカル・ユニットについてのログイン不成立の応答をする場合に、ログイン要求を受けた前記特定装置に対して、該特定装置がそのロジカル・ユニットに再度ログイン要求を出す場合の再要求タイミングを指示することを特徴とする被ログイン装置。

【請求項 7】 請求項 1 ないし請求項 6 のうちの任意の 1 つに記載の被ログイン装置において、

複数の前記特定装置は前記通信路を介してログイン要求を同時に出せないよう調整されていることを特徴とする被ログイン装置。

【請求項 8】 請求項 1 ないし請求項 6 のうちの任意の 1 つに記載の被ログイン装置において、

前記通信路は IEEE 1394 バスから成ること特徴とする被ログイン装置。

【請求項 9】 請求項 1 ないし請求項 6 のうちの任意の 1 つに記載の被ログイン装置において、

前記特定装置と、SBP-2 プロトコルで通信を行なうことを特徴とする被ログイン装置。

【請求項 10】 或る特定装置に所定の通信路を介してログインし得るログイン装置であって、

前記特定装置に対してログイン要求を出した後に、前記特定装置から、ログイン不成立の応答を受けると共に、再要求タイミングの指示を受けた場合に、指示された該再要求タイミングで前記特定装置に対して再度ログイン要求を出すログイン要求手段

を備えることを特徴とするログイン装置。

【請求項 11】 請求項 10 に記載のログイン装置において、他の装置とは前記通信路を介してログイン要求を同時に出せないよう調整されていることを特徴とするログイン装置。

【請求項 12】 請求項 10 に記載のログイン装置において、前記通信路は IEEE 1394 バスから成ること特徴とするログイン装置。

【請求項 13】 請求項 10 に記載のログイン装置において、前記特定装置と、SBP-2 プロトコルで通信を行なうことを特徴とするログイン装置。

【請求項 14】 請求項 1 ないし請求項 9 のうちの任意の 1 つに記載の被ログイン装置と請求項 10 ないし請求項 13 のうちの任意の 1 つに記載のログイン装置とを備える装置間通信システム。

【請求項 15】 複数のログイン装置から所定の通信路を介して、被ログイン装置の備える 1 つ以上のロジカル・ユニットにログインするためのログイン制御方法であって、

(a) 前記被ログイン装置が、或るログイン装置から或るロジカル・ユニットに対するログイン要求を受けた場合において、既にそのロジカル・ユニットにログインしている前記ログイン装置の数が前記ロジカル・ユニットについて予め定められたログイン許容数に達している際には、前記被ログイン装置が、ログイン要求を受けた前記ログイン装置に対して、前記ロジカル・ユニットについてのログイン不成立の応答をする工程と、

(b) 前記被ログイン装置が、前記ロジカル・ユニットについてのログイン不成立の応答をする場合に、ログイン要求を受けた前記ログイン装置に対して、該ログイン装置がそのロジカル・ユニットに再度ログイン要求を出す場合の再要求タイミングを指示する工程と、

(c) 前記ログイン装置が、前記被ログイン装置から、前記ロジカル・ユニットについてのログイン不成立の応答を受けると共に、再要求タイミングの指示を受けた場合に、指示された該再要求タイミングで前記被ログイン装置における前記ロジカル・ユニットに対して再度ログイン要求を出す工程と、

を備えることを特徴とするログイン制御方法。

【請求項 1 6】 請求項 1 5 に記載のログイン制御方法において、

前記工程 (b) は、前記被ログイン装置が、ログイン要求を受けた前記特定装置について、最初のログイン要求を受けた順番に、その優先順位を設定し、設定した前記優先順位の高いログイン装置ほど前記再要求タイミングを短くするように、前記再要求タイミングを決定する工程を含むことを特徴とするログイン制御方法。

【請求項 1 7】 複数の特定装置から所定の通信路を介してログインされ得るコンピュータに被ログイン処理を行なわせるためのコンピュータプログラムを記録したコンピュータ読み取り可能な記録媒体であって、

(a) 或る特定装置からログイン要求を受けた場合に、既にログインしている前記特定装置の数が予め定められたログイン許容数に達している際には、ログイン要求を受けた前記特定装置に対して、ログイン不成立の応答をする工程と、

(b) ログイン不成立の応答をする場合に、ログイン要求を受けた前記特定装置に対して、該特定装置が再度ログイン要求を出す場合の再要求タイミングを指示する工程と

を前記コンピュータに実行させるためのコンピュータプログラムを記録した記録媒体。

【請求項 1 8】 請求項 1 7 に記載の記録媒体において、

前記工程 (b) は、ログイン要求を受けた前記特定装置について、最初のログイン要求を受けた順番に、その優先順位を設定し、設定した前記優先順位の高い特定装置ほど前記再要求タイミングを短くするように、前記再要求タイミングを決定する工程を含むことを特徴とする記録媒体。

【請求項 1 9】 或る特定装置に所定の通信路を介してログインし得るコンピュータにログイン処理を行なわせるためのコンピュータプログラムを記録した

コンピュータ読み取り可能な記録媒体であって、

前記特定装置に対してログイン要求を出した後に、前記特定装置から、ログイン不成立の応答を受けると共に、再要求タイミングの指示を受けた場合に、指示された該再要求タイミングで前記特定装置に対して再度ログイン要求を出す工程を前記コンピュータに実行させるためのコンピュータプログラムを記録した記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、複数のログイン装置から所定の通信路を介して被ログイン装置にログインするための技術に関するものである。特に、装置間をIEEE 1394バスで接続されたシステムにおいて、装置間の通信をSBP (Serial Bus Protocol) - 2プロトコルで行なう場合に用いて好適な技術に関するものである。

【0002】

【従来の技術】

IEEE 1394は、IEEEによって提唱されたデジタルデータをやり取りするためのシリアルバス規格であって、「IEEE STD.1394-1995 IEEE Standard for a High Performance serial Bus」として規定されている。

【0003】

一方、SBP-2プロトコル（以下、単にSBP-2と略す）は、米国のANSI X3T10が標準化を推進するプロトコルであって、IEEE 1394規格に準拠して設計されたトランスポート層のプロトコルである。

【0004】

さて、装置間をIEEE 1394バスで接続されたシステムにおいて、装置間の通信をSBP-2で行なう場合、共有メモリを持つ装置をイニシエータ (Initiator) と呼び、持たない方の装置をターゲット (target) と呼ぶ。

【0005】

ターゲットは、それぞれ、サービスを提供する1つ以上のロジカル・ユニット (Logical Unit) を備えている。イニシエータは、そのサービスを提供するロジ

カル・ユニットを利用するために、ターゲット内のそのロジカル・ユニットにログイン (Login) して、そのロジカル・ユニットを利用する権利を獲得する必要がある。具体的には、イニシエータがターゲットのそのロジカル・ユニットにログイン要求を出し、そのロジカル・ユニットからログイン成立の応答を受けることによって、行なわれる。

【0006】

その後、イニシエータは、そのロジカル・ユニットの利用が完了したら、ターゲット内のそのロジカル・ユニットからログアウト (Logout) して、そのロジカル・ユニットを利用する権利を破棄する。具体的には、イニシエータが、そのロジカル・ユニットにログアウト要求を出し、そのロジカル・ユニットからログアウト成功の応答を受けることによって、なされる。

【0007】

ところで、ターゲットの備えるロジカル・ユニットは、それぞれ、ロジカル・ユニット毎に、同時にログインしていることを許すイニシエータの数 (ログイン許容数) が、設計の段階で予め定められている。従って、ターゲットのロジカル・ユニットにおいて、既にログインされているイニシエータの数が、そのロジカル・ユニットのログイン許容数に達している場合には、そのロジカル・ユニットは、その後、未だログインされていない他のイニシエータからログイン要求を受けても、そのイニシエータに対してはログイン不成立の応答を行なうことになる。

【0008】

一方、そのイニシエータは、そのロジカル・ユニットからログイン不成立の応答を受けると、それ以降、ログイン成立の応答を受け取るまで、一定時間毎に繰り返し、そのロジカル・ユニットにログイン要求を出す。その後、そのロジカル・ユニットにログインされているイニシエータのうち、何れかのイニシエータがログアウトすると、そのロジカル・ユニットにログインされているイニシエータの数はログイン許容数に満たなくなるので、そのロジカル・ユニットは、繰り返しログイン要求を出していた上記イニシエータから、次にログイン要求を受けると、そのイニシエータに対しログイン成立の応答を返す。これにより、繰り返しログイン要求を出していた上記イニシエータはそのロジカル・ユニットにログイ

ンして、そのロジカル・ユニットを利用する権利を獲得することができる。

【0009】

【発明が解決しようとする課題】

しかしながら、従来においては、次のような問題があった。今、ターゲットの或るロジカル・ユニットに対して、ログインされているイニシエータの数がログイン許容数に達している場合に、第1のイニシエータがログイン要求を出し、その後、第2のイニシエータがそのロジカル・ユニットに対してログイン要求を出し始めた場合を考えてみる。このとき、第1のイニシエータと第2のイニシエータは、それぞれ、異なるタイミングで一定時間毎に、そのロジカル・ユニットに対してログイン要求を繰り返し出すことになる。その後、そのロジカル・ユニットに既にログインされているイニシエータの何れかがログアウトすると、その後に来る、第1のイニシエータのログイン要求を出すタイミングが、第2のイニシエータのログイン要求を出すタイミングより早い場合には、先にログイン要求を出し始めた第1のイニシエータがログインすることになるため、何ら問題はない。しかし、第1のイニシエータのログイン要求を出すタイミングが、第2のイニシエータのログイン要求を出すタイミングより遅い場合には、後からログイン要求を出し始めた第2のイニシエータが第1のイニシエータを追い越してログインしてしまうという問題があった。

【0010】

しかも、第2のイニシエータ以外にもログイン要求を出し始めるイニシエータが現れれば、第1のイニシエータは、さらに、それらイニシエータによっても追い越されて先にそのロジカル・ユニットにログインされてしまう可能性もあり、そのような場合に、ロジカル・ユニットになかなかログインできず、そのロジカル・ユニットを利用することができないという問題があった。

【0011】

従って、本発明の目的は、上記した従来技術の問題点を解決し、ターゲットに対し最先にログイン要求を出したイニシエータが、最先にログインすることを概ね保証することができる技術を提供することにある。

【0012】

【課題を解決するための手段およびその作用・効果】

上記した目的の少なくとも一部を達成するために、本発明の被ログイン装置は、複数の特定装置から所定の通信路を介してログインされ得る被ログイン装置であって、

或る特定装置からログイン要求を受けた場合に、既にログインしている前記特定装置の数が予め定められたログイン許容数に達している際には、ログイン要求を受けた前記特定装置に対して、ログイン不成立の応答をする応答手段と、

ログイン不成立の応答をする場合に、ログイン要求を受けた前記特定装置に対して、該特定装置が再度ログイン要求を出す場合の再要求タイミングを指示する再要求タイミング指示手段と、

を備えることを要旨とする。

【0013】

このように、本発明の被ログイン装置においては、或る特定装置からログイン要求を受けた場合に、既にログインしている特定装置の数が予め定められたログイン許容数に達している際には、応答手段が、ログイン要求を受けた特定装置に対して、ログイン不成立の応答をする。そして、再要求タイミング指示手段が、そのログイン不成立の応答をする特定装置に対して、その特定装置が再度ログイン要求を出す場合の再要求タイミングを指示する。

【0014】

従って、本発明の被ログイン装置においては、再要求タイミング指示手段が、特定装置に対して再要求タイミングを指示することによって、ログイン要求を受けている特定装置毎に、異なる再要求タイミングを指示することができるため、最先にログイン要求を出した特定装置に対して、その特定装置が最先にログインし得るような再要求タイミングを指示することも可能である。従って、そのような再要求タイミングを指示した場合には、最先にログイン要求を出した特定装置が、最先にログインすることを概ね保証することができる。

【0015】

なお、本明細書において、ログインとは、例えば、被ログイン装置を利用する権利を獲得するための手続きを言う。

【0016】

本発明の被ログイン装置において、前記再要求タイミング指示手段は、ログイン要求を受けた前記特定装置について、その優先順位を設定する優先順位設定手段と、

設定された前記優先順位に従って、前記特定装置についての前記再要求タイミングを決定する再要求タイミング決定手段と、

を備えることが好ましい。

【0017】

このように構成することによって、ログイン要求を受けている特定装置に対し、それぞれ、その優先順位に応じた再要求タイミングを指示することができるようになる。

【0018】

本発明の被ログイン装置において、前記優先順位設定手段は、

最初のログイン要求を受けた順番に、前記特定装置の優先順位を設定することが好ましい。

【0019】

このように優先順位を設定することにより、最先にログイン要求を出した特定装置が最も優先順位が高くなるように設定されるため、この優先順位に応じて再要求タイミングを決定する際に、その特定装置に対して、最先にログインし得るような再要求タイミングを決定することも可能となる。

【0020】

本発明の被ログイン装置において、前記再要求タイミング決定手段は、設定された前記優先順位の高い特定装置ほど、前記再要求タイミングを短くするように決定することが好ましい。

【0021】

このように再要求タイミングを決定することにより、優先順位の高い特定装置ほど、指示する再要求タイミングは短くなる。従って、最も優先順位が高い特定装置は、他の特定装置に比較して、再度ログイン要求を行なう場合の再要求タイミングが最も短くなるため、単位時間当たりに出すログイン要求の数は最も多く

なる。よって、既にログインされている特定装置の何れかがログアウトした場合に、その後、最も優先順位が高い特定装置が最先にログイン要求を出して最先にログインする可能性が、最も高くなる。

【0022】

また、優先順位を最初のログイン要求を受けた順番に設定した場合は、最も優先順位が高い特定装置は、最先にログイン要求を出した特定装置となるので、最先にログイン要求を出した特定装置が最先にログインする可能性が、最も高くなる。

【0023】

本発明の被ログイン装置において、前記再要求タイミング指示手段は、前記再要求タイミングを、再度ログイン要求を出すまでの時間間隔を表す時定数として指示するようにしても良い。

【0024】

本発明の被ログイン装置において、

前記特定装置から互いに独立にログインされ得る1つ以上のロジカル・ユニットを備え、

前記応答手段は、或る特定装置から或るロジカル・ユニットにログイン要求を受けた場合に、既にそのロジカル・ユニットにログインしている前記特定装置の数がそのロジカル・ユニットについて予め定められたログイン許容数に達している際には、ログイン要求を受けた前記特定装置に対して、そのロジカル・ユニットについてのログイン不成立の応答をすると共に、

前記再要求タイミング指示手段は、そのロジカル・ユニットについてのログイン不成立の応答をする場合に、ログイン要求を受けた前記特定装置に対して、該特定装置がそのロジカル・ユニットに再度ログイン要求を出す場合の再要求タイミングを指示することが好ましい。

【0025】

このように、被ログイン装置が、独立にログインされ得る1つ以上のロジカル・ユニットを備える場合にも、特定装置に対して、ロジカル・ユニット毎に再要求タイミングを指示することができる。

【0026】

本発明のログイン装置は、或る特定装置に所定の通信路を介してログインし得るログイン装置であって、

前記特定装置に対してログイン要求を出した後に、前記特定装置から、ログイン不成立の応答を受けると共に、再要求タイミングの指示を受けた場合に、指示された該再要求タイミングで前記特定装置に対して再度ログイン要求を出すログイン要求手段

を備えることを要旨とする。

【0027】

このように、本発明のログイン装置においては、ログイン要求手段が、特定装置から指示された再要求タイミングで、特定装置に対して再度ログイン要求を行なう。従って、ログイン装置が特定装置に最先にログイン要求を出したログイン装置である場合に、特定装置から、最先にログインし得るような再要求タイミングが指示されれば、その指示された再要求タイミングでログイン要求を出すことにより、最先にログインすることが概ね保証され得る。

【0028】

本発明の被ログイン装置において、複数の前記特定装置は前記通信路を介してログイン要求を同時に出せないよう調整されていることが好ましい。また、本発明のログイン装置において、他の装置とは前記通信路を介してログイン要求を同時に出せないよう調整されていることが好ましい。

【0029】

これにより、一般的なネットワークで用いられるキャリア検出やコリジョン検出といったリソースの排他制御の方式を不要とすることができる。

【0030】

本発明の被ログイン装置またはログイン装置において、前記通信路はIEEE 1394バスから成ることが好ましい。

【0031】

このように通信路をIEEE 1394バスによって構成することにより、高速通信を利用して、特定装置との間でデータのやり取りを行なうことができる。

【 0 0 3 2 】

通信路が I E E E 1 3 9 4 バスから成る本発明の被ログイン装置またはログイン装置において、特定装置と、 S B P - 2 プロトコルで通信を行なうことが好ましい。

【 0 0 3 3 】

S B P - 2 プロトコルは、 I E E E 1 3 9 4 規格のバスシステムの利用を前提に設計されたプロトコルであって、ハードディスク装置や C D - R O M 装置などに広く普及したトランスポート層のプロトコルであるため、商用パーソナルコンピュータのオペレーティング・システムに、標準で実装される場合が多いからである。

【 0 0 3 4 】

本発明の装置間通信システムは、上記した本発明の被ログイン装置と本発明のログイン装置とを備えることを要旨とする。

【 0 0 3 5 】

本発明の装置間通信システムによれば、被ログイン装置に対し最先にログイン要求を出したログイン装置が、最先にログインすることを概ね保証することができる。

【 0 0 3 6 】

本発明のログイン制御方法は、複数のログイン装置から所定の通信路を介して、被ログイン装置の備える 1 つ以上のロジカル・ユニットにログインするためのログイン制御方法であって、

(a) 前記被ログイン装置が、或るログイン装置から或るロジカル・ユニットに対するログイン要求を受けた場合において、既にそのロジカル・ユニットにログインしている前記ログイン装置の数が前記ロジカル・ユニットについて予め定められたログイン許容数に達している際には、前記被ログイン装置が、ログイン要求を受けた前記ログイン装置に対して、前記ロジカル・ユニットについてのログイン不成立の応答をする工程と、

(b) 前記被ログイン装置が、前記ロジカル・ユニットについてのログイン不成立の応答をする場合に、ログイン要求を受けた前記ログイン装置に対して、該

ログイン装置がそのロジカル・ユニットに再度ログイン要求を出す場合の再要求タイミングを指示する工程と、

(c) 前記ログイン装置が、前記被ログイン装置から、前記ロジカル・ユニットについてのログイン不成立の応答を受けると共に、再要求タイミングの指示を受けた場合に、指示された該再要求タイミングで前記被ログイン装置における前記ロジカル・ユニットに対して再度ログイン要求を出す工程と、

を備えることを要旨とする。

【0037】

本発明のログイン制御方法においては、被ログイン装置が、ログイン要求を受けたログイン装置に対して再要求タイミングを指示することによって、ログイン要求を受けているログイン装置毎に、異なる再要求タイミングを指示することができるため、最先にログイン要求を出したログイン装置に対して、そのログイン装置が最先にログインし得るような再要求タイミングを指示することも可能である。従って、そのような再要求タイミングを指示した場合には、最先にログイン要求を出したログイン装置が、最先にログインすることを概ね保証することができる。

【0038】

本発明のログイン制御方法において、前記工程(b)は、前記被ログイン装置が、ログイン要求を受けた前記特定装置について、最初のログイン要求を受けた順番に、その優先順位を設定し、設定した前記優先順位の高いログイン装置ほど前記再要求タイミングを短くするように、前記再要求タイミングを決定する工程を含むことが好ましい。

【0039】

このように再要求タイミングを決定することにより、最初のログイン要求を先に受けたログイン装置ほど、優先順位が高くなり、指示する再要求タイミングは短くなる。従って、最先にログイン要求を出したログイン装置は、他のログイン装置に比較して、再度ログイン要求を行なう場合の再要求タイミングが最も短くなるため、単位時間当たりに出すログイン要求の回数は最も多くなる。よって、既にログインされているログイン装置の何れかがログアウトした場合に、その後

、そのログイン装置が最先にログイン要求を出して最先にログインする可能性が、最も高くなる。

【0040】

本発明の記録媒体は、複数の特定装置から所定の通信路を介してログインされ得るコンピュータに被ログイン処理を行なわせるためのコンピュータプログラムを記録したコンピュータ読み取り可能な記録媒体であって、

(a) 或る特定装置からログイン要求を受けた場合に、既にログインしている前記特定装置の数が予め定められたログイン許容数に達している際には、ログイン要求を受けた前記特定装置に対して、ログイン不成立の応答をする工程と、

(b) ログイン不成立の応答をする場合に、ログイン要求を受けた前記特定装置に対して、該特定装置が再度ログイン要求を出す場合の再要求タイミングを指示する工程と、を前記コンピュータに実行させるためのコンピュータプログラムを記録したことを要旨とする。

【0041】

本明細書において、コンピュータには、パーソナルコンピュータや、モバイルコンピュータ、情報処理端末装置や、ワークステーションなど、種々のコンピュータが含まれる他、実質的にコンピュータ機能を有するプリンタや複写機やインターフェイス切り換え器などの各種周辺機器や、同じくコンピュータ機能を有するオーディオセットやビデオセットなど各種家庭用機器や業務用機器なども含まれる。但し、後述する【発明の実施の形態】の欄においては、「コンピュータ」という語を、パーソナルコンピュータに代表されるような狭義のコンピュータという意味で用いる場合もある。

【0042】

上記のようなコンピュータプログラムをコンピュータによって実行させると、上記した本発明の被ログイン装置における応答手段や再生要求タイミング指示手段と同様な機能を実現することができるため、上記した被ログイン装置と同様の効果を奏することが可能となる。

【0043】

本発明の記録媒体において、前記工程 (b) は、ログイン要求を受けた前記特

定装置について、最初のログイン要求を受けた順番に、その優先順位を設定し、設定した前記優先順位の高い特定装置ほど前記再要求タイミングを短くするように、前記再要求タイミングを決定する工程を含むことが好ましい。

【0044】

このように構成することにより、最先にログイン要求を出した特定装置は再要求タイミングが最も短くなるため、単位時間当たりに出すログイン要求の回数は最も多くなる。従って、既にログインされている特定装置の何れかがログアウトした場合に、その後、その特定装置が最先にログイン要求を出して、最先にログインする可能性が最も高くなる。

【0045】

本発明の記録媒体は、或る特定装置に所定の通信路を介してログインし得るコンピュータにログイン処理を行なわせるためのコンピュータプログラムを記録したコンピュータ読み取り可能な記録媒体であって、

前記特定装置に対してログイン要求を出した後に、前記特定装置から、ログイン不成立の応答を受けると共に、再要求タイミングの指示を受けた場合に、指示された該再要求タイミングで前記特定装置に対して再度ログイン要求を出す工程を前記コンピュータに実行させるためのコンピュータプログラムを記録したことを要旨とする。

【0046】

このようなコンピュータプログラムをコンピュータによって実行させると、上記した本発明のログイン装置におけるログイン要求手段と同様な機能を実現することができるため、上記したログイン装置と同様の効果を奏することが可能となる。

【0047】

なお、本発明は、被ログイン装置、ログイン装置、装置間通信システム、ログイン制御方法、それら装置を構築するためのコンピュータプログラム、そのコンピュータプログラムを記録した記録媒体、そのコンピュータプログラムを含み搬送波内に具現化されたデータ信号、等の種々の態様で実現することができる。

【0048】

【発明の実施の形態】

以下、本発明の実施の形態を実施例に基づいて説明するが、その前に、IEEE 1394バス及びSBP-2について、その概要を説明する。

【0049】

(A) IEEE 1394バス及びSBP-2の概要

IEEE 1394バスは、64ビットのアドレス空間を持つプロセッサバスをシリアルライズして同一信号線上にのせたものであり、転送速度は現状では400MHzであるが、将来的には最高で3.2GHzを想定している。即ち、IEEE 1394バスは、コンピュータのマザーボード上に見られるような通常のプロセッサバスと基本的には変わらないが、巨大なメモリ空間を1本の信号線によってサポートしていることになる。

【0050】

また、IEEE 1394バスはバスであるが、1つのプロセッサ専用として用いられるわけではなく、複数のプロセッサをつなげることができる。これはイメージとして、1つのバスの持っているメモリ空間を複数のプロセッサに割り当てた場合に相当する。即ち、複数のプロセッサが1つの信号経路を共有できるので、結果としてネットワークに非常に似た状況が実現される。

【0051】

このように、IEEE 1394バスでは、1つのバスに複数のコンピュータを接続することが可能であるから、そのバスにさらに、例えばプリンタを接続すれば、そのプリンタを複数のコンピュータで共有することも可能になる。これはあたかも小規模なネットワークとして見ることができる。

【0052】

しかしながら、一般的なネットワークとは異なり、IEEE 1394バスはプロセッサバスの設計を踏襲しているので、基本的には、個々のプロセッサが出す要求を衝突させずに管理することが可能である。このメカニズムをバスアービトレーションという。即ち、データ転送中のIEEE 1394バス上では、このバスアービトレーションによって、1つのノード（即ち、この場合は1つのコンピュータ）だけがデータ転送を実行することが保証されている。言い換えると、或

るノードがIEEE 1394バスを占有してしまうと、他のノードは同一サイクルタイム（125 μ s）内ではそのバスを使用することができないような制御が働く。このため、一般的なネットワークで用いられるキャリア検出やコリジョン（パケットの衝突）検出といったリソースの排他制御の方式は、IEEE 1394バスでは不必要となっている。しかしながら、IEEE 1394バスの場合、このようなコリジョン検出を行なわないため、リソースの競合は上位層プロトコルなし、応用プログラムの責任で行なわなくてはならないという問題がある。

【0053】

一方、SBP-2の設計上の特徴は、いわゆるコネクションベースであることである。即ち、或るSBP-2のターゲットに対して、これを利用したいホストデバイスは、SBP-2のイニシエータとして設計される必要がある。そして、イニシエータは、ターゲットに対してログインを要求し、そのログインが成立した場合のみ、ターゲットを利用する権利が獲得される（なお、正確には、前述したように、イニシエータは、ターゲットの備えるロジカル・ユニット毎にログインするのであるが、ここでは、説明を簡単にするために、単にターゲットにログインするものとして説明する。）。従って、仮に、複数のイニシエータに対して、1つのターゲットがあった場合でも、ログインの許されたイニシエータのみがそのターゲットを利用するようにすることによって、複数のイニシエータによって、そのターゲットを共用することができる。

【0054】

なお、SBP-2のそのものの仕様は、1つのターゲットに対して複数のイニシエータによるログインを禁止してはいないので、ターゲットさえ、そのように設計することが可能であれば、同時に複数のイニシエータによるログインを許しても構わない。なお、同時にログインしていることを許すイニシエータの数（即ち、ログイン許容数）は、イニシエータを設計する段階で、設計上の限界に応じて規定される。しかしながら、或るターゲットのログイン許容数がN個である場合、それを上回るM（ $>N$ ）個のイニシエータが、そのターゲットに対しログインを要求する場合は往々にしてある。そのような場合、ターゲットは、N個のイ

ニシエータまではログインを許すが、それ以降は、同時にログインしているイニシエータの数がN個を下回らない限り、即ち、ログインしている何れかのイニシエータがログアウトしない限り、ログイン要求がいくつ来ても、すべて拒絶する。

【0055】

ところで、SBP-2において、イニシエータとターゲットはIEEE1394バスで接続された形になっている。前述したように、IEEE1394バスはバスであるので、ターゲットはイニシエータのメモリに対しリード/ライトすることができる。そこで、前述したように、SBP-2では、共有メモリを有している装置をイニシエータと呼び、共有メモリを有しない装置をターゲットと呼んでいる。即ち、両方の装置が共有メモリを共に持つことは無駄であるので、一方の装置が共有メモリを持つようにしている。従って、イニシエータは自己の有するメモリが共有メモリとなっているので、ローカルアクセスでデータをリード/ライトすることができる。これに対して、ターゲットはイニシエータの有するメモリを共有メモリとして使用しているので、リモートアクセスでデータをリード/ライトする必要がある。なお、装置に対するイニシエータ、ターゲットの役割付けは、装置を設計する際の最初段階で行なわれる。通常は、コンピュータなどのホストデバイスがイニシエータになって、プリンタやスキャナなどのデバイスがターゲットになる場合が多い。この役割付けは固定的なもので、原則として途中で変わることはない。但し、設計によっては、1つの装置において、スイッチ等によりイニシエータ、ターゲットの役割付けの切り換えができるようにしても構わない。

【0056】

さて、前述したようにイニシエータがターゲットにログインすると、SBP-2では、次のようにしてトランザクションを行なう。

【0057】

まず、イニシエータが、自己の有する共有メモリ上にORB (Operation Request Block) という特別なデータ構造を設け、ここにターゲットに使ってもらいたいデータを置く。例えば、イニシエータがコンピュータであり、ターゲットが

プリンタである場合には、イニシエータであるコンピュータが、ターゲットであるプリンタに印刷してもらいたい画像データを共有メモリ空間のいずれかに置き、次にこのアドレスのポインタを予め用意したORBの所定位置に格納する。このため、イニシエータは、ターゲットの都合によらず、自己の達成できる最大の速度で処理可能である。一方のターゲットは、イニシエータの有する共有メモリ上に置かれたORBを順次取り込んで、その内容を消費するように動く。即ち、ターゲットであるプリンタは、イニシエータである画像データを印刷することにより自己のメモリに空きができ次第、次の画像データを消費するので、いつも休むことなく印刷を続けることができ、ターゲットも最大の効率でジョブの処理ができる。

【0058】

従来の一般的なネットワークでは、イニシエータがターゲットの状態をポーリングして、ターゲットが要求を受け付けるレディ状態になったときに、イニシエータがコマンドを送るというようなプッシュスタイルの制御を行なっていたが、SBP-2では、イニシエータは、自己のメモリ空間にターゲットのための命令やデータを置いておき、ターゲットが自分の処理可能なタイミングでその命令やデータを取り込んで処理するというプルスタイルの制御を行なう。これにより、SBP-2では、全体のトランザクションの効率が高くなる。

【0059】

(B) 実施例の構成

以上のことを踏まえて、本発明の一実施例について図1を用いて説明する。図1は本発明の一実施例としてのイニシエータやターゲットを備えた装置間通信システムを示すブロック図である。本システムでは、各装置間を図1に示すようにIEEE1394バスB1で接続しており、また、装置間の通信をSBP-2プロトコルで行なっている。従って、IEEE1394バスB1に接続された各装置は、共有メモリを有するイニシエータI0～I3と、共有メモリを有していないターゲットT1と、に分かれている。なお、本実施例においては、イニシエータI0～I3はそれぞれコンピュータであり、ターゲットT1はプリンタであるものとする。

【0060】

各イニシエータI0～I3は、それぞれ、コントローラC0～C3と、共有メモリSM0～SM3と、を備えている。

【0061】

一方、ターゲットT1は、1つのロジカル・ユニットLUN1を備えている。前述したように、イニシエータは、ターゲットの備えるロジカル・ユニット毎に、ログインすることが可能であるが、本実施例では、ターゲットT1の有するロジカル・ユニットは1つであるので、イニシエータI0～I3は、ターゲットT1に対し、このロジカル・ユニットLUN1のみにログインすることが可能となる。

【0062】

また、前述したように、ターゲットの備えるロジカル・ユニットは、それぞれ、ロジカル・ユニット毎に、同時にログインしていることを許すイニシエータの数（ログイン許容数）が、設計の段階で予め定められている。本実施例では、説明を簡単にするために、ターゲットT1の備えるロジカル・ユニットLUN1のログイン許容数は「1」としている。ロジカル・ユニットLUN1は、イニシエータの何れか1つがログインすると、それ以降、他のイニシエータからログイン要求を受けても、拒否することになる。

【0063】

また、ターゲットT1の備えるロジカル・ユニットLUN1は、イニシエータからのログイン／ログアウト要求を受け、それに対する応答を行なうマネジメント・エージェントME1と、ログインされたイニシエータとの間でデータのやり取りを行なうフェッチ・エージェントFE1と、を備えている。

【0064】

図2は図1に示すイニシエータを構成するコンピュータとターゲットを構成するプリンタのハードウェア構成の概要を示すブロック図である。

【0065】

図2に示すように、ハードウェアの構成として、イニシエータを構成するコンピュータ100は、その内部に、コンピュータプログラムに従って種々の処理や

制御を行なうためのCPU102と、上記コンピュータプログラムを記憶したり、処理中に得られたデータなどを一時的に記憶したりするためのメモリ104と、CPU102と各種周辺装置やIEEE1394バスB1との間でデータなどのやり取りを行なうためのI/O部106と、各種データを格納するためのハードディスク装置108と、CD-ROMドライブ装置110と、を備えている。また、その外部には、ユーザからの指示などを入力するためのキーボード112やマウス114と、データなどを表示するためのモニタ116と、を備えている。

【0066】

このうち、CPU102が、メモリ104に格納された所望のコンピュータプログラムを読み出して実行することにより、図1に示したコントローラとして機能する。また、図1に示した共用メモリには、メモリ104の一部が割り当てられる。

【0067】

一方、ターゲットを構成するプリンタ200は、その内部に、コンピュータプログラムに従って種々の処理や制御を行なうためのCPU202と、上記コンピュータプログラムを記憶したり、処理中に得られたデータなどを一時的に記憶したりするためのメモリ204と、CPU202と他の構成要素やIEEE1394バスB1との間でデータなどのやり取りを行なうためのI/O部206と、CPU202からの指示に従って、印刷用紙に画像等の印刷を行なうプリンタ機構部208と、印刷データを一時的に格納するための入力バッファ210と、を備える。

【0068】

このうち、CPU202が、メモリ204に格納された所望のコンピュータプログラムを読み出して実行することにより、図1に示したロジカル・ユニットとして機能する。

【0069】

本実施例では、コンピュータ100のメモリ104に格納されているコンピュータプログラムは、記録媒体であるCD-ROM111に記録された形態で提供

され、CD-ROMドライブ装置 110 により読み取られることによって、コンピュータ 100 内に取り込まれる。取り込まれたコンピュータプログラムは、各々のハードディスク装置 108 に転送され、その後、起動時などにメモリ 104 に転送される。あるいは、読み取られたコンピュータプログラムは、各々のハードディスク装置 108 を介さず、直接、メモリ 104 に転送するようにしても良い。

【0070】

一方、プリンタ 200 の方は、メモリ 204 の一部が ROM で構成されており、プリンタ用のコンピュータプログラムはその ROM に格納されている。従って、このコンピュータプログラムは ROM のチップに記録された形態で提供される。また、このような ROM を用いる代わりに、メモリ 104 の一部を書き換え可能な不揮発性メモリで構成すると共に、プリンタ用のコンピュータプログラムを CD-ROM 111 に記録された形態で提供し、そのコンピュータプログラムをコンピュータ 100 の CD-ROM ドライブ装置 110 によって読み取って、図示せざるネットワーク等を介してプリンタ 200 に転送し、上記の不揮発性メモリに書き込むようにしても良い。

【0071】

このように、本実施例では、コンピュータプログラムをコンピュータ読み取り可能に記録する「記録媒体」として CD-ROM や ROM を利用することを述べたが、その他にも、フレキシブルディスクや光磁気ディスク、IC カード、ROM カートリッジ、パンチカード、バーコードなどの符号が印刷された印刷物、コンピュータの内部記憶装置（RAM などのメモリ）および外部記憶装置等の、コンピュータが読取り可能な種々の媒体を利用できる。

【0072】

また、コンピュータプログラムは、このような記録媒体に記録された形態での提供の他、ネットワークを介して、コンピュータプログラムを供給するプログラムサーバ（図示せず）にアクセスし、プログラムサーバからコンピュータ 100 やプリンタ 200 内に取り込むようにしても良い。

【0073】

また、上記コンピュータプログラムの一部は、オペレーティングシステムプログラムによって構成するようにしても良い。

【0074】

(C) 実施例の動作

それでは、本実施例の動作について、図3、図4及び図5を参照しながら説明する。

【0075】

図3は図1に示すイニシエータにおけるログイン処理の処理手順を示すフローチャートであり、図4は図1に示すターゲットにおけるログイン処理の処理手順を示すフローチャートである。また、図5は図1に示す各イニシエータとターゲットとの間の要求・応答の受け渡しを示すタイミングチャートである。図5において、縦軸はそれぞれ時間軸であり、左から順に、イニシエータI0、ターゲットT1、イニシエータI1、イニシエータI2、イニシエータI3における要求・応答の受け渡しのタイミングを示している。

【0076】

なお、各イニシエータI0～I3、ターゲットT1はそれぞれIEEE1394バスB1で接続されているため、各イニシエータI0～I3からターゲットT1に対して要求を送出する場合、前述したバスアービトレーションによって、1つのイニシエータが要求を送出している間はIEEE1394バスB1はそのイニシエータに占有されるため、他のイニシエータから同時に要求が送出されることはない。

【0077】

図5に示すように、まず、イニシエータI0がターゲットT1にログイン要求を出す場合、イニシエータI0のコントローラC0は、図3に示すように、ターゲットT1に対しログイン・パケットを送出する(ステップS102)。このログイン・パケットには、ターゲットT1にログインを要求するためのログイン要求と、イニシエータI0のGUIDと、が含まれている。ここで、GUIDとは、グローバル・ユニーク・アイデンティフィケーション(Global Unique Identification)の略で、各イニシエータ、ターゲットに固有の識別符号のことである。

このGUIDは、イニシエータ、ターゲットを構成する装置がそれぞれ製造される段階で、世界中で唯一の識別符号として与えられる。本実施例では、イニシエータI0のGUIDは「I0」とし、同じく、イニシエータI1のGUIDは「I1」、イニシエータI2のGUIDは「I2」、イニシエータI3のGUIDは「I3」としている。従って、イニシエータI0が送出したログイン・パケットに含まれるGUIDは、図5に示すように「I0」となる。

【0078】

一方、ターゲットT1のマネージメント・エージェントME1は、図4に示すように、ログイン・パケットを受け取るまで待機している（ステップS204）。そして、マネージメント・エージェントME1は、パケットを受け取ると、そのパケットにログイン要求が含まれているかどうかを判定し、ログイン要求が含まれていれば、そのパケットをログイン・パケットと判断して、次のステップS206の処理に進む。

【0079】

ステップS206では、マネージメント・エージェントME1は、受け取ったログイン・パケットに含まれるGUIDを読み込む。前述したように、イニシエータI0が送出したログイン・パケットを受け取った場合は、読み込んだGUIDは「I0」であるので、マネージメント・エージェントME1は、そのGUIDから、そのログイン・パケットはイニシエータI0から送出されたものであると判断する。

【0080】

ところで、マネージメント・エージェントME1は、ログイン要求を受けたイニシエータのGUIDを登録するための待ち行列を有している。図6はマネージメント・エージェントME1の有する待ち行列の一例を示す説明図である。図6において、(a)～(h)は図5に示すタイミングチャートの時系列に対応した順番に並んでいる。

【0081】

待ち行列に対するGUIDの登録は、イニシエータからの最初のログイン要求を受けた順に、図6(a)に示す番地n=1～7に行なわれる。従って、番地n

= 1 ~ 7 は、それぞれ、最初のログイン要求を受けた順位に対応し、本実施例では、これをイニシエータの優先順位とする。また、番地 $n = 0$ には、ロジカル・ユニット LUN 1 にログインされているイニシエータの GUID が登録される。前述したように、ロジカル・ユニット LUN 1 のログイン許容数は、予め「1」と定められており、同時にログインされるイニシエータの数は 1 つであるので、1 つの番地 $n = 0$ のみが、ログインされているイニシエータのために割り当てられている。

【0082】

そこで、マネージメント・エージェント ME 1 は、ステップ S 206 で読み込んだ GUID が、上記した待ち行列に既に登録されているかどうかを判定する（ステップ S 208）。この場合、イニシエータ I 0 からのログイン要求は今回が初めてである（即ち、最初のログイン要求である）ので、読み込んだ GUID である「I 0」は、図 6（a）に示したように、未だ待ち行列には登録されておらず、従って、ステップ S 218 の処理に進む。

【0083】

ステップ S 218 では、マネージメント・エージェント ME 1 は、読み込んだ GUID を待ち行列の末尾に登録する。この場合、待ち行列の番地 $n = 1 \sim 7$ には未だ何も登録されていないので、マネージメント・エージェント ME 1 は、読み込んだ GUID である「I 0」を、図 6（b）に示すように待ち行列の番地 $n = 1$ に登録することになる。

【0084】

次に、マネージメント・エージェント ME 1 は、ロジカル・ユニット LUN 1 にログインされているイニシエータの数 p を読み込む（ステップ S 210）。上記したように、イニシエータがログインされると、待ち行列の番地 $n = 0$ には、必ず、そのイニシエータの GUID が登録されるので、マネージメント・エージェント ME 1 は、その番地 $n = 0$ をチェックし、既に GUID が登録されていれば、ログインされているイニシエータの数 p として「1」を読み込み、未だ登録されていないならば、 p として「0」を読み込む。なお、この場合は、図 6（b）に示すように、待ち行列の番地 $n = 0$ には未だ GUID が登録されていないので

、pとして「0」を読み込むことになる。

【0085】

続いて、マネージメント・エージェントME1は、読み込んだイニシエータの数pが、予め定められているロジカル・ユニットLUN1のログイン許容数「1」と等しいか否かを判定する（ステップS212）。即ち、これにより、マネージメント・エージェントME1は、ロジカル・ユニットLUN1に既にログインされているイニシエータの数pが、そのロジカル・ユニットLUN1のログイン許容数「1」に達しているかどうかを判定する。

【0086】

判定の結果、ログインされているイニシエータの数pがログイン許容数「1」に達している場合には、もはやこれ以上、ロジカル・ユニットLUN1はイニシエータをログインさせることができないので、ステップS213以降の処理に移る。なお、ステップS213以降の処理については後述する。

【0087】

一方、判定の結果、ログインされているイニシエータの数pがログイン許容数「1」に達していない場合には、ロジカル・ユニットLUN1はそのログイン要求のあったイニシエータをログインさせることができるので、ステップS220以降の処理に移る。

【0088】

即ち、マネージメント・エージェントME1は、まず、ログイン要求を受けたイニシエータI0に対して、ステータスをログイン成立を表す「ログイン・サクセスフル」に設定したステータス・パケットを送出して（ステップS220）、図5に示すようにログインが成立した旨の応答を行なう。

【0089】

次に、マネージメント・エージェントME1は、待ち行列に登録されているGUID全体を移動させる（ステップS222）。具体的には、この場合、待ち行列に登録されているGUIDは、図6（b）に示すように番地n=1に登録されているイニシエータI0のGUIDだけであるので、このGUIDを、図6（c）に示すようにログインされたこと示す番地n=0に移動させる。

【0090】

こうして、ステップS222の処理が終了すると、マネージメント・エージェントME1は、再び、次のログイン・パケットを受け取るまで待機している（ステップS204）。

【0091】

一方、イニシエータI0のコントローラC0は、図3に示すように、ステップS102でログイン・パケットを送出した後、ステータス・パケットを受け取るまで待機している（ステップS104）。その後、コントローラC0は、ターゲットT1からのステータス・パケットを受け取ると、そのパケットに含まれるステータスを読み込む（ステップS106）。

【0092】

そして、コントローラC0は、読み込んだステータスが、ログイン不成立を表す「ログイン・エラー」であるかどうかを判定する（ステップS108）。判定の結果、「ログイン・エラー」であった場合には、ログイン不成立であったので、再度、ターゲットT1に対してログイン要求を出すために、コントローラC0は、ステップS110以降の処理を行なう。なお、このステップS110以降の処理については、後述する。

【0093】

一方、判定の結果、ステータスが「ログイン・エラー」でなく、「ログイン・サクセスフル」であった場合には、ログインが成立したので、コントローラC0は、図3に示す一連のログイン処理を終了する。

【0094】

以上のようにして、イニシエータI0がターゲットT1のロジカル・ユニットLUN1にログインすると、それ以降、イニシエータI0におけるコントローラC0は、ターゲットT1におけるフェッチ・エージェントFE1との間で、実際のデータなどのやり取りを行なう。

【0095】

こうして、イニシエータI0がターゲットT1のロジカル・ユニットLUN1にログインした後に、図5に示すように、次に、別のイニシエータI1がターゲ

ットT1にログイン要求を出す場合、次のような処理となる。

【0096】

即ち、イニシエータI1のコントローラC1が、まず、図3に示すように、ターゲットT1に対し、ログイン要求とイニシエータI1のGUIDを含んだログイン・パケットを送出する（ステップS102）。この場合、ログイン・パケットに含まれるGUIDは、図5に示すように「I1」となる。

【0097】

これに対し、マネージメント・エージェントME1は、図4に示すように、ログイン・パケットを受け取ると、受け取ったログイン・パケットに含まれるGUIDを読み込んで（ステップS206）、そのGUIDである「I1」から、そのログイン・パケットはイニシエータI1から送出されたものであると判断する。

【0098】

次に、マネージメント・エージェントME1は、読み込んだGUIDが待ち行列に既に登録されているかどうかを判定する（ステップS208）。この場合、イニシエータI1からのログイン要求は今回が初めてであるので、読み込んだGUIDである「I1」は、図6（c）に示したように、未だ待ち行列には登録されていない。従って、マネージメント・エージェントME1は、読み込んだGUIDを待ち行列の末尾に登録する（ステップS218）。この場合、待ち行列の番地 $n=0$ にはイニシエータI0のGUIDである「I0」が登録されているが、番地 $n=1\sim7$ には未だ何も登録されていないので、マネージメント・エージェントME1は、今回読み込んだGUIDである「I1」を、図6（d）に示すように番地 $n=1$ に登録することになる。

【0099】

次に、マネージメント・エージェントME1は、ロジカル・ユニットLUN1にログインされているイニシエータの数 p を読み込む（ステップS210）。この場合は、図6（d）に示すように、待ち行列の番地 $n=0$ には、既にイニシエータI0のGUIDが登録されているので、 p として「1」を読み込むことになる。

【0100】

次に、マネージメント・エージェントME1は、読み込んだイニシエータの数pが、ログイン許容数「1」と等しいか否かを判定する（ステップS212）。この場合、pは「1」であり、ログイン許容数「1」と等しいので、マネージメント・エージェントME1は、ロジカル・ユニットLUN1に既にログインされているイニシエータの数pが、ログイン許容数「1」に達しており、もはやこれ以上、ロジカル・ユニットLUN1はイニシエータをログインさせることができないと判断して、ステップS213以降の処理に移る。

【0101】

即ち、マネージメント・エージェントME1は、上記した待ち行列から、対象となっているイニシエータのGUIDの優先順位nを読み込む（ステップS213）。この場合、待ち行列の状態は図6（d）に示すごとくであり、対象となっているイニシエータI1のGUIDは番地n=1に登録されており、従って、その優先順位nは「1」であるので、この「1」を読み込むことになる。

【0102】

次に、マネージメント・エージェントME1は、予め用意されている時定数テーブルから、ステップS213で読み込んだ優先順位nに対応した時定数を読み込む（ステップS214）。

【0103】

ここで、時定数とは、ログイン要求を出したイニシエータが、ログイン不成立の場合に、再度、ログイン要求を出すまでの時間を言う。本実施例では、これから述べるように、ログインが不成立であった場合に、ターゲットは、ログイン要求を出したイニシエータに対して、再度ログイン要求を出す場合の再要求タイミングを指示する。このとき、その再生要求タイミングとして、上記時定数を与える。

【0104】

本実施例においては、ログイン要求を出しているイニシエータの優先順位（即ち、上記した待ち行列における優先順位）nと、この時定数と、の間に所定の関係を設けており、マネージメント・エージェントME1は、その関係を表す時定

数テーブルを予め用意している。図7はマネージメント・エージェントME1の有する時定数テーブルの一例を示す説明図である。

【0105】

そこで、ステップS214では、マネージメント・エージェントME1が、この時定数テーブルから優先順位nに対応した時定数を読み込む。この場合、ステップS213で読み込んだ優先順位nは「1」であるので、図7に示す時定数テーブルから、時定数として、n=1に対応した「t1」を読み込むことになる。

【0106】

続いて、マネージメント・エージェントME1は、ログイン要求を受けたイニシエータI1に対して、ステータスをログイン不成立を表す「ログイン・エラー」に設定したステータス・パケットを送出して（ステップS216）、図5に示すように、ログインが不成立であった旨の応答を行なう。このとき、マネージメント・エージェントME1は、上記したステータスの他に、図5に示すように、ステップS214で読み込んだ時定数である「t1」もステータス・パケットに含めて送出手。これにより、マネージメント・エージェントME1は、ログイン要求を受けたイニシエータI1に対して、そのイニシエータI1が、再度、ログイン要求を出す場合の再要求タイミングを指示することになる。

【0107】

こうして、ステップS216の処理が終了すると、マネージメント・エージェントME1は、再び、次のログイン・パケットを受け取るまで待機している（ステップS204）。

【0108】

一方、イニシエータI1のコントローラC1は、図3に示すように、ターゲットT1からのステータス・パケットを受け取ると、そのパケットに含まれるステータスを読み込み（ステップS106）、そのステータスが、ログイン不成立を表す「ログイン・エラー」であるかどうかを判定する（ステップS108）。この場合、読み込んだステータスは「ログイン・エラー」であって、ログイン不成立であるので、コントローラC1は、再度、ターゲットT1に対してログイン要求を出すために、ステップS110以降の処理を行なう。

【0109】

即ち、コントローラC1は、まず、受け取ったステータス・パケットに含まれる時定数を読み込み（ステップS110）、次に、ステップS102においてターゲットT1に対してログイン・パケットを送出してから時間として、読み込んだ時定数の示す時間が経過したか否かを判定する（ステップS112）。この場合、読み込んだ時定数は「t1」であるので、前回ログイン要求を出してから、時間t1経過したかどうかを判定することになる。判定の結果、経過してなければ、経過するまで待機するが、経過していれば、再びステップS102の処理に戻って、コントローラC1は、再度、ターゲットT1に対してログイン・パケットを送出する（ステップS102）。この結果、図5に示すように、イニシエータI1からターゲットT1に対しては、1回目のログイン要求を送出してから時間t1経過した後に、2回目のログイン要求を送出することになる。このようにして、コントローラC1は、ターゲットT1から時定数として指示された再要求タイミングに基づいて、再度、ターゲットT1に対してログイン要求を出す。

【0110】

一方、イニシエータI1が1回目のログイン要求を出して2回目のログイン要求を出すまでの間に、さらに別のイニシエータI2、I3がターゲットT1にログイン要求を出す場合、それらの処理はイニシエータI1の場合と同様な処理となる。

【0111】

即ち、イニシエータI2のコントローラC2とイニシエータI3のコントローラC3が、それぞれ、図5に示すようなタイミングで、ログイン要求と各々のGUIDを含めて、ターゲットT1に対しログイン・パケットを送出する（ステップS102）と、ターゲットT1のマネージメント・エージェントME1は、図5に示すようなタイミングで、各々のログイン・パケットを受け取って（ステップS204）、それぞれに含まれるGUIDを読み込み（ステップS206）、待ち行列に登録されているか否かを判定する（ステップS208）。この場合、イニシエータI2、I3からのログイン要求は今回がそれぞれ初めてであるので、読み込んだGUIDは、何れも、図6（d）に示したように、未だ待ち行列に

は登録されていない。従って、マネージメント・エージェントME1は、読み込んだGUIDを待ち行列の末尾に順次登録する（ステップS218）。具体的には、図6（d）に示すように、待ち行列の番地 $n=1$ には既にイニシエータI1のGUIDである「I1」が登録されているので、イニシエータI2のログイン要求を受けた時点では、読み込んだイニシエータI2のGUIDである「I2」を、図6（e）に示すように番地 $n=2$ に登録し、その後、イニシエータI3のログイン要求を受けた時点では、読み込んだイニシエータI3のGUIDである「I3」を、図6（f）に示すように番地 $n=3$ に登録することになる。

【0112】

続いて、上記したいずれの時点においても、ロジカル・ユニットLUN1には既にイニシエータI0がログインされているので、マネージメント・エージェントME1は、ログインされているイニシエータの数 p がログイン許容数「1」に達していると判断して（ステップS210、S212）、待ち行列から、対象となっているイニシエータのGUIDの優先順位 n を読み込む（ステップS213）。イニシエータI2のログイン要求を受けた時点では、待ち行列の状態は図6（e）に示すごとくであり、イニシエータI2のGUIDは番地 $n=2$ に登録されており、その優先順位 n として「2」を読み込むことになる。また、イニシエータI3のログイン要求を受けた時点では、図6（f）のごとくであり、イニシエータI2のGUIDは番地 $n=3$ に登録されており、その優先順位 n として「3」を読み込むことになる。従って、ステップS214において、マネージメント・エージェントME1が、図7に示す時定数テーブルから、これら優先順位 n に対応した時定数を読み込むと、イニシエータI2については、時定数として、 $n=2$ に対応した「 t_2 」を読み込み、イニシエータI3については、 $n=3$ に対応した「 t_3 」を読み込むことになる。

【0113】

本実施例においては、図7に示すように、優先順位 n に対応した時定数は、それぞれ、優先順位 n が高くなるほど短くなり、低くなるほど長くなっている。具体的には、優先順位 $n=1$ に対応した時定数「 t_1 」を基準にすると、 $n=2$ に対応した時定数「 t_2 」は「 t_1 」の2倍となっており、 $n=3$ に対応した時定

数「 t_3 」は「 t_1 」の4倍となっており、 $n=4$ に対応した時定数「 t_4 」は「 t_1 」の8倍となっている。即ち、 $n=i$ に対応した時定数「 t_i 」は「 t_1 」の 2^i 倍になっている。但し、時定数が余りに長くなりすぎても、効果が期待できないので、優先順位 n が低い $n=7$ については、その時定数「 t_7 」を、 $n=6$ に対応した時定数「 t_6 」（即ち、「 t_1 」の32倍）と等しくしている。

【0114】

続いて、マネージメント・エージェントME1は、ログイン要求を受けたイニシエータI2、I3に対して、図5に示すようなタイミングで、それぞれ、「ログイン・エラー」と時定数を含むステータス・パケットを送出する（ステップS216）。すると、イニシエータI2のコントローラC2とイニシエータI3のコントローラC3は、それぞれ、図5に示すようなタイミングで、そのステータス・パケットを受け取って（ステップS104）、ステータスを読み込み（ステップS106）、そのステータスが「ログイン・エラー」であるかどうかを判定する（ステップS108）。この場合、ステータスは「ログイン・エラー」であるので、続いて、コントローラC2、C3は、ステータス・パケットから時定数を読み込む（ステップS110）。この場合、コントローラC2の読み込む時定数は「 t_2 」であり、コントローラC3の読み込む時定数は「 t_3 」であるので、コントローラC2は、前回ログイン要求を出してから時間 t_2 経過した後に、再度ターゲットT1に対してログイン・パケットを送出し、コントローラC3は、前回ログイン要求を出してから時間 t_3 経過した後に、再度ターゲットT1に対してログイン・パケットを送出する（ステップS112、S102）。この結果、図5に示すように、イニシエータI2からターゲットT1に対しては、1回目のログイン要求を送出してから時間 t_2 経過した後に、2回目のログイン要求を送出することになり、イニシエータI3からターゲットT1に対しては、1回目のログイン要求を送出してから時間 t_3 経過した後に、2回目のログイン要求を送出することになる。

【0115】

なお、以上のようにして、イニシエータI1～I3がそれぞれ2回目以降のログイン要求をターゲットT1に対して出した場合、ターゲットT1のマネジメ

ント・エージェントME1は、何れの場合も、ログイン・パケットから読み込んだGUIDが既に待ち行列に登録されているので、ステップS208の判定により、ステップS218の処理を介することなく、ステップS210の処理に進むことになる。

【0116】

また、本実施例の場合、図5に示すように、最後のイニシエータI3が1回目のログイン要求を出した後、ログインしていたイニシエータI0がログアウトするまでの間は、イニシエータI1～I3以外のイニシエータがターゲットT1に対して新たにログイン要求を出すことはないので、マネージメント・エージェントME1の備える待ち行列の状態は、図6(f)の状態から変化はない。従って、待ち行列に登録されているイニシエータI1～I3のGUIDの優先順位にも変化はないので、イニシエータI1～I3から2回目以降のログイン要求を受けた場合にも、マネージメント・エージェントME1は、ログイン要求を受けたイニシエータに対して、前回送出した時定数と同様の時定数をステータス・パケットに含めて送出する。従って、それを受けたイニシエータI1～I3は、それぞれ、図5に示すように、前回と同様の時定数の表すタイミングで、再度のログイン要求をターゲットT1に対して送出することになる。

【0117】

以上のようにして、本実施例においては、既にロジカル・ユニットLUN1にイニシエータI0がログインされている場合において、他のイニシエータI1～I3については、最初のログイン要求を先に出したイニシエータほど、ログイン要求を出してから次のログイン要求を出すまでの時間（即ち、時定数）が短くなるため、単位時間当たりに出すログイン要求の数は多くなる。具体的には、イニシエータI1, I2, I3の時定数の比は、図7から明らかなように1:2:4であるので、単位時間当たりのログイン要求数の比は逆に4:2:1となる。

【0118】

そこで、ログインしているイニシエータI0が任意のタイミングでログアウトすると、次に、ロジカル・ユニットLUN1にログインするのは、ログアウト後に、最も早くログイン要求を出したイニシエータである。従って、単位時間当た

りに出すログイン要求の数が多いイニシエータほど、上記のタイミングで（即ち、ログアウト後、最も早く）ログイン要求を出す可能性は高くなるので、次にロジカル・ユニット LUN 1 にログインする確率は、イニシエータ I 1 が最も高くなる。従って、本実施例によれば、最先にログイン要求を出したイニシエータが、最先にログインすることを概ね保証することができる。

【0119】

さて、その後、イニシエータ I 0 が、図 5 に示すようなタイミングでターゲット T 1 にログアウト要求を出す場合、次のような処理となる。

【0120】

即ち、イニシエータ I 0 におけるコントローラ C 0 と、ターゲット T 1 におけるフェッチ・エージェント F E 1 と、の間で行なわれていたデータのやり取りが完了すると、コントローラ C 0 は、ターゲット T 1 に対してログアウト・パケットを送出する。このログアウト・パケットには、図 5 に示すように、ターゲット T 1 にログアウトを要求するためのログアウト要求と、イニシエータ I 0 の G U I D と、が含まれている。

【0121】

一方、ターゲット T 1 では、図 8 に示すようなログアウト処理を行なう。図 8 は図 1 に示すターゲットにおけるログアウト処理の処理手順を示すフローチャートである。

【0122】

即ち、ターゲット T 1 のマネージメント・エージェント M E 1 は、図 8 に示すように、ログアウト・パケットを受け取るまで待機しており（ステップ S 3 0 2）、パケットを受け取ると、そのパケットにログアウト要求が含まれているかどうかを判定し、ログアウト要求が含まれていれば、そのパケットをログアウト・パケットと判断して、次のステップ S 3 0 4 の処理に進む。

【0123】

ステップ S 3 0 4 では、マネージメント・エージェント M E 1 は、受け取ったログアウト・パケットに含まれる G U I D を読み込む。この場合、読み込んだ G U I D は「I 0」であるので、マネージメント・エージェント M E 1 は、その G

UIDから、そのログアウト・パケットはイニシエータI0から送出されたものであると判断する。

【0124】

続いて、マネージメント・エージェントME1は、待ち行列の番地 $n=0$ に登録されているGUID、即ち、ログインされているイニシエータのGUIDと、ステップS304で読み込んだGUIDと、を比較して、同じGUIDであることを確認した上で、その登録されているGUIDを待ち行列から削除する（ステップS306）。即ち、この場合、待ち行列の状態は図6（f）に示すごとくになっており、番地 $n=0$ には、ログインされているイニシエータのGUIDとして、「I0」が登録されている。一方、ステップS304で読み込んだGUIDも「I0」であり、登録されているGUIDと同じIDなので、マネージメント・エージェントME1は、図6（g）に示すように、番地 $n=0$ に登録されている「I0」を削除する。

【0125】

そして、マネージメント・エージェントME1は、ログアウト要求を受けたイニシエータI0に対して、ステータスをログアウト成立を表す「ログアウト・サクセスフル」に設定したステータス・パケットを送出して（ステップS308）、図5に示すようにログアウトが成立した旨の応答を行なう。

【0126】

こうして、ステップS308の処理が終了すると、マネージメント・エージェントME1は、再び、次のログアウト・パケットを受け取るまで待機している（ステップS302）。

【0127】

以上によって、ログインしていたイニシエータI0は、ターゲットT1のロジカル・ユニットLUN1からログアウトしたことになる。

【0128】

さて、イニシエータI0がロジカル・ユニットLUN1からログアウトした後、図5に示すようなタイミングで、イニシエータI1がターゲットT1にログイン要求を出す場合、次のような処理となる。

【0129】

この場合、イニシエータI1とターゲットT1の処理は、イニシエータI0がターゲットT1のロジカル・ユニットLUN1にログインしたときの処理とほぼ同様となる。但し、ターゲットT1におけるログイン処理のうち、ステップS210以降の処理は以下ようになる。

【0130】

即ち、マネージメント・エージェントME1は、ロジカル・ユニットLUN1にログインされているイニシエータの数pを読み込む（ステップS210）。この場合、図6（g）に示すように、待ち行列の番地n=0に登録されていたGUIDは、前述したとおり削除されているので、pとして「0」を読み込むことになる。

【0131】

次に、マネージメント・エージェントME1は、読み込んだイニシエータの数pがログイン許容数「1」と等しいか否かを判定する（ステップS212）が、この場合、ログインされているイニシエータの数pはログイン許容数「1」に達していないので、マネージメント・エージェントME1は、まず、ログイン要求を受けたイニシエータI1に対して、ステータスをログイン成立を表す「ログイン・サクセスフル」に設定したステータス・パケットを送出して（ステップS220）、図5に示すようにログインが成立した旨の応答を行なう。続いて、マネージメント・エージェントME1は、待ち行列に登録されているGUID全体を移動させる（ステップS222）。具体的には、この場合、図6（g）に示すように、待ち行列の番地n=1にはイニシエータI1のGUIDが、番地n=2にはイニシエータI2のGUIDが、番地n=3にはイニシエータI3のGUIDが、それぞれ登録されている。従って、まず、番地n=1に登録されているイニシエータI1のGUIDを空いている番地n=0に移動させる。これにより、ログイン要求を受けたイニシエータI1のGUIDは、ログインされたこと示す番地n=0に移動することになる。次に、番地n=2, 3にそれぞれ登録されているイニシエータI2, I3のGUIDを、それぞれ、上位に1つずつ移動させる。この結果、待ち行列の状態は図6（h）に示すごとくになる。

【0132】

以上のようにして、イニシエータI0がログアウトした後に、最も早くログイン要求を出したイニシエータI1が、次に、ターゲットT1のロジカル・ユニットLUN1にログインしたことになる。

【0133】

続いて、イニシエータI1がログインした後に、図5に示すようなタイミングで、イニシエータI2、I3がそれぞれターゲットT1にログイン要求を出した場合、次のような処理となる。

【0134】

この場合、イニシエータI2、I3とターゲットT1の処理は、イニシエータI2、I3が前回ターゲットT1に対してログイン要求を出したときの処理とほぼ同様となるが、次の点が異なってくる。

【0135】

即ち、ターゲットT1におけるログイン処理のうち、ステップS213以降において、マネージメント・エージェントME1は、まず、待ち行列から、対象となっているイニシエータのGUIDの優先順位nを読み込む（ステップS213）が、このとき、待ち行列の状態は図6（h）に示すごとくになっている。従って、イニシエータI2について優先順位nを読み込む場合、イニシエータI2のGUIDは前回とは異なり、番地n=1に登録されているので、その優先順位nとして「1」を読み込むことになる。また、イニシエータI3について優先順位nを読み込む場合も、イニシエータI3のGUIDは前回と違って番地n=2に登録されているので、その優先順位nとして「2」を読み込むことになる。従って、イニシエータI2、I3共に、優先順位nがそれぞれ前回より1つつ上がっている。

【0136】

続いて、マネージメント・エージェントME1が、図7に示す時定数テーブルから、これら優先順位nに対応した時定数を読み込む（ステップS214）と、イニシエータI2については、時定数として、n=1に対応した「t1」を読み込み、イニシエータI3については、n=2に対応した「t2」を読み込むこと

になる。従って、イニシエータ I 2 についての時定数は、前回「t 2」であったのが、今回「t 1」となり、イニシエータ I 3 についての時定数は、前回「t 3」であったのが、今回「t 2」となるため、何れの場合も、前回より $1/2$ 短くなる。

【0137】

その後、マネージメント・エージェント ME 1 が、ログイン要求を受けたイニシエータ I 2, I 3 に対して、図 5 に示すようなタイミングで、それぞれ、「ログイン・エラー」と時定数を含んだステータス・パケットを送出する（ステップ S 216）と、イニシエータ I 2 のコントローラ C 2 とイニシエータ I 3 のコントローラ C 3 は、それぞれ、そのステータス・パケットを受け取って（ステップ S 104）、そのステータス・パケットから時定数を読み込む（ステップ S 110）。この場合、コントローラ C 2 の読み込む時定数は「t 1」であり、コントローラ C 3 の読み込む時定数は「t 2」であるので、コントローラ C 2 は、前回ログイン要求を出してから時間 t 1 経過した後に、再度ターゲット T 1 に対してログイン・パケットを送出し、コントローラ C 3 は、前回ログイン要求を出してから時間 t 2 経過した後に、再度ターゲット T 1 に対してログイン・パケットを送出する（ステップ S 112, S 102）。

【0138】

この結果、イニシエータ I 1 がログインされた後は、イニシエータ I 2, I 3 がターゲット T 1 に対してそれぞれログイン要求を出す場合、ログイン要求を出してから次のログイン要求を出すまでの時間（即ち、時定数）が、イニシエータ I 1 がログインされる前よりも、それぞれ、 $1/2$ 短くなるため、単位時間当たりに出すログイン要求の数は、それぞれ、2 倍に増えることになる。従って、ログインしているイニシエータ I 1 が任意のタイミングでログアウトしても、次にログインするイニシエータは、早期にログイン要求を出してログインすることができる。

【0139】

しかしながら、このように、イニシエータ I 1 がログインされる前よりも、イニシエータ I 2, I 3 の時定数がそれぞれ短くなり、ログイン要求数がそれぞれ

多くなっても、イニシエータ I 2 と I 3 との関係においては、最初のログイン要求を先に出したイニシエータ I 2 の方が、イニシエータ I 3 に比較して、時定数が短く、単位時間当たりに出すログイン要求の数が多いことに変わりはない。従って、ログインしているイニシエータ I 1 が任意のタイミングでログアウトした場合に、その後、最も早くログイン要求を出す可能性があるのは、単位時間当たりに出すログイン要求の数が多いイニシエータ I 2 であるので、次にロジカル・ユニット LUN 1 にログインする確率は、イニシエータ I 2 が最も高くなる。

【0140】

(D) 補足

ところで、上記した説明においては、各イニシエータ I 0 ~ I 3 からターゲット T 1 へはログイン・パケットやログアウト・パケットを送出し、ターゲット T 1 からイニシエータ I 0 ~ I 3 へはステータス・パケットを送出するとして説明したが、これらの動作について、さらに詳細な説明を加える。

【0141】

図 1 に示したとおり、各イニシエータ I 0 ~ I 3 は、それぞれ、共有メモリ S M 0 ~ S M 3 を備えている。そこで、イニシエータ I 0 ~ I 3 がターゲット T 1 に対してログイン・パケットやログアウト・パケットを送出する場合は、まず、イニシエータ I 0 ~ I 3 が、ターゲット T 1 に送りたいログイン・パケットやログアウト・パケットのデータを、図 1 に示すようなマネージメント ORB と呼ばれる特別なデータ構造として、自己の有する共有メモリ S M 0 ~ S M 3 上に置く。次に、イニシエータ I 0 ~ I 3 は、ターゲット T 1 のマネージメント・エージェント M E 1 に対して、IEEE 1394 バス B 1 を介して、そのデータを置いた共有メモリ上の番地を示すポインタを送り出す。一方、ターゲット T 1 のマネージメント・エージェント M E 1 は、このポインタを受け取ると、IEEE 1394 バス B 1 を介して、ポインタを送り出したイニシエータの共有メモリにアクセスして、ポインタの示す番地に置かれているマネージメント ORB を取り込むことにより、ログイン・パケットやログアウト・パケットのデータを受け取る。こうして、イニシエータ I 0 ~ I 3 からターゲット T 1 へログイン・パケットやログアウト・パケットが送出される。

【0142】

一方、その応答として、ターゲットT1がイニシエータI0～I3に対してステータス・パケットを送出する場合は、次のようになる。即ち、イニシエータI0～I3から受け取ったログイン・パケットやログアウト・パケットには、要求やGUIDの他に、イニシエータが応答を受け取るために用いる、共有メモリ上の番地を示すポインタが含まれている。そこで、ターゲットT1のマネージメント・エージェントME1は、ログイン・パケットやログアウト・パケットを受け取ると、それに含まれている上記のポインタを読み取り、次に、IEEE1394バスB1を介して、イニシエータの共有メモリにアクセスして、読み取ったポインタの示すSTATUS_FIFOに、ステータス・パケットのデータを書き込む。これによって、ターゲットT1からイニシエータI0～I3へ、ステータス・パケットがリモートライトされる。

【0143】

なお、イニシエータがターゲットT1のロジカル・ユニットLUN1にログインした後に、イニシエータと、ターゲットT1のフェッチ・エージェントFE1と、の間で行なわれるデータのやり取りも、上記と同様の方法で行なわれる。但し、フェッチ・エージェントFE1との間でデータのやり取りを行なう場合、イニシエータが、自己の有する共有メモリ上に置くデータは、上記したマネージメントORBではなく、図1に示すようなノーマルORBと呼ばれるデータ構造として置くことになる。

【0144】

さて、本実施例では、前述したとおり、イニシエータI0がログインしている場合に、そのイニシエータI0が任意のタイミングでログアウトすると、その後、最も早くログイン要求を出す可能性が高いのは、単位時間当たりに出しているログイン要求の数が最も多いイニシエータI1である。そのため、図5に示した例では、イニシエータI0がログアウトした後に、イニシエータI1が最も早くターゲットT1にログイン要求を出す場合について説明した。しかしながら、イニシエータI0がログアウトするタイミングによっては、イニシエータI1以外のイニシエータI2やI3が、イニシエータI0がログアウトした後に、最も早

くターゲットT1にログイン要求を出す可能性がないわけではない。例えば、イニシエータI1が前回ログイン要求を出してから今回のログイン要求を出すまでの間に、イニシエータI2やI3がログイン要求を出す場合において、イニシエータI1が前回ログイン要求を出してからイニシエータI2やI3がログイン要求を出すまでの間に、イニシエータI0がログアウトしてしまうと、ログアウト後に、イニシエータI2またはI3が最も早くターゲットT1に対しログイン要求を出すことになる。

【0145】

このような場合、ログアウト後に、最も早くログイン要求を出したイニシエータI2またはI3が、ターゲットT1のロジカル・ユニットLUN1にログインすることになる。即ち、最初のログイン要求を先に出したイニシエータI1は、後からログイン要求を出したイニシエータI2またはI3に追い越されて、先にログインされてしまうことになる。

【0146】

さて、このような場合において、ターゲットT1におけるログイン処理のうち、ステップS222の処理) 即ち、マネージメント・エージェントME1が待ち行列に登録されているGUID全体を移動させる処理) は、次のようになる。今、イニシエータI0がログアウトした後に、イニシエータI2が最も早くターゲットT1にログイン要求を出し、ログインする場合を例にとる。

【0147】

待ち行列の状態は図6(g)に示すごとくになっており、待ち行列の番地n=1にはイニシエータI1のGUIDが、番地n=2にはイニシエータI2のGUIDが、番地n=3にはイニシエータI3のGUIDが、それぞれ登録されている。そこで、マネージメント・エージェントME1は、ログイン要求を受けたイニシエータI2のGUIDを、番地n=2から、ログインされたことを示す番地n=0に移動させる。そして、イニシエータI3のGUIDを番地n=3から空席となった番地n=2に移動させて、上位に1つ繰り上げる。なお、イニシエータI1のGUIDは、そのままとする。

【0148】

以上のような処理を行なった場合、最初のログイン要求を先に出したイニシエータ I 1 は、後からログイン要求を出したイニシエータ I 2 にログインは譲るものの、待ち行列におけるイニシエータ I 1 の GUID の優先順位 n には変更はないので、イニシエータ I 2 がログインした後も、ログイン要求を出してから次のログイン要求を出すまでの時間（即ち、時定数）は前と変わらず最も短いままであり、単位時間あたりに出すログイン要求の数も前と変わらず最も多いままである。そのため、ログインしたイニシエータ I 2 が任意のタイミングでログアウトした後に、最も早くログイン要求を出してログインする可能性が一番高いのは、イニシエータ I 1 であることに変わりはない。

【0149】

しかしながら、以上のような処理を行なった場合、次のような不都合を生じる可能性がある。即ち、例えば、最初のログイン要求を先に出したイニシエータ I 1 が、繰り返し出すはずのログイン要求を、何らかの事情（例えば、イニシエータ I 1 が IEEE 1394 バス B 1 から外された場合など）によって出さなくなってしまった場合に、上記のような処理を行なうと、待ち行列には、番地 $n = 1$ （即ち、優先順位 n が「1」）に登録されているイニシエータ I 1 の GUID がそのまま残っているため、番地 $n = 3$ から番地 $n = 2$ に移動したイニシエータ I 3 の優先順位 n は「2」であり、イニシエータ I 1 がターゲット T 1 に対しログイン要求を出さなくなったにも関わらず、イニシエータ 3 についての時定数としては、優先順位 $n = 2$ に対応した「 t_2 」が用いられることになり、ターゲット T 1 に対してログイン要求を出すタイミングが無意味に長くなってしまう。

【0150】

そこで、これを改善するために、ステップ S 222 の処理について、上記した処理に代えて、次のような処理を行なうようにしても良い。

【0151】

待ち行列の状態が図 6（g）に示すごとくになっている場合に、マネージメント・エージェント ME 1 が、ログイン要求を受けたイニシエータ I 2 の GUID を、番地 $n = 2$ から、ログインされたことを示す番地 $n = 0$ に移動させると共に、そのイニシエータ I 2 の GUID よりも優先順位 n が上位にあるイニシエータ

のGUID（即ち、イニシエータI1のGUID）を待ち行列から削除する。そして、イニシエータI2のGUIDよりも優先順位nが下位にあるイニシエータI3のGUIDを、番地n=3から空席となった番地n=1に移動させて、上位に2つ繰り上げる。このような処理を行なうことによって、上記した不都合を改善することができる。

【0152】

また、他の方法としては、マネージメント・エージェントME1が、各イニシエータから繰り返し受けるログイン要求のタイミングをそれぞれ検知し、イニシエータからのログイン要求が途絶えた場合に、所定の時間待っても、再度ログイン要求が来ないときには、そのイニシエータのGUIDを待ち行列から削除するようにしても良い。

【0153】

なお、本発明は上記した実施例や実施形態に限られるものではなく、その要旨を逸脱しない範囲において種々の態様にて実施することが可能である。

【0154】

上記した実施例においては、図1に示すように、IEEE1394バスB1に接続されているイニシエータは4つであり、ターゲットは1つであったが、本発明はこれらの数に限定されるものではない。従って、例えば、ターゲットが2つ以上あっても構わず、各ターゲットにおいて上記したような処理が行なわれれば良い。

【0155】

また、上記した実施例においては、ターゲットT1の備えるロジカル・ユニットは1つであったが、2つ以上であっても良く、その場合には、各イニシエータは、各ロジカル・ユニット毎に、それぞれ、独立にログインすることができる。また、ロジカル・ユニットLUN1のログイン許容数は「1」としたが、2以上としても良く、その場合には、図4に示したターゲットのログイン処理のうち、ステップS212における判定を、その数に対応させる必要がある。

【0156】

また、上記した実施例においては、イニシエータはコンピュータ、ターゲット

はプリンタとしたが、本発明はこれに何ら限定されるものではない。例えば、イニシエータをコンピュータ以外に、例えば、スキャナやその他デバイスとしても良い。また、ターゲットもプリンタ以外のデバイスとしても良い。或いは、ターゲットをいわゆるインターフェイス切り換え器とし、ターゲットにIEEE1394バスから既存のインターフェイスへの切り換えという役割だけを持たせ、その先にスキャナやプリンタをつなげるようにしても良い。

【図面の簡単な説明】

【図1】

本発明の一実施例としての装置間通信システムを示すブロック図である。

【図2】

図1に示すイニシエータを構成するコンピュータとターゲットを構成するプリンタのハードウェア構成の概要を示すブロック図である。

【図3】

図1に示すイニシエータにおけるログイン処理の処理手順を示すフローチャートである。

【図4】

図1に示すターゲットにおけるログイン処理の処理手順を示すフローチャートである。

【図5】

図1に示す各イニシエータとターゲットとの間の要求・応答の受け渡しを示すタイミングチャートである。

【図6】

マネージメント・エージェントME1の有する待ち行列の一例を示す説明図である。

【図7】

マネージメント・エージェントME1の有する時定数テーブルの一例を示す説明図である。

【図8】

図1に示すターゲットにおけるログアウト処理の処理手順を示すフローチャートである。

トである。

【符号の説明】

100…コンピュータ

102…CPU

104…メモリ

106…I/O部

108…ハードディスク装置

110…CD-ROMドライブ装置

111…CD-ROM

112…キーボード

114…マウス

116…モニタ

200…プリンタ

202…CPU

204…メモリ

206…I/O部

208…プリンタ機構部

210…入力バッファ

B1…IEEE1394バス

C0～C3…コントローラ

FE1…フェッチ・エージェント

I0～I3…イニシエータ

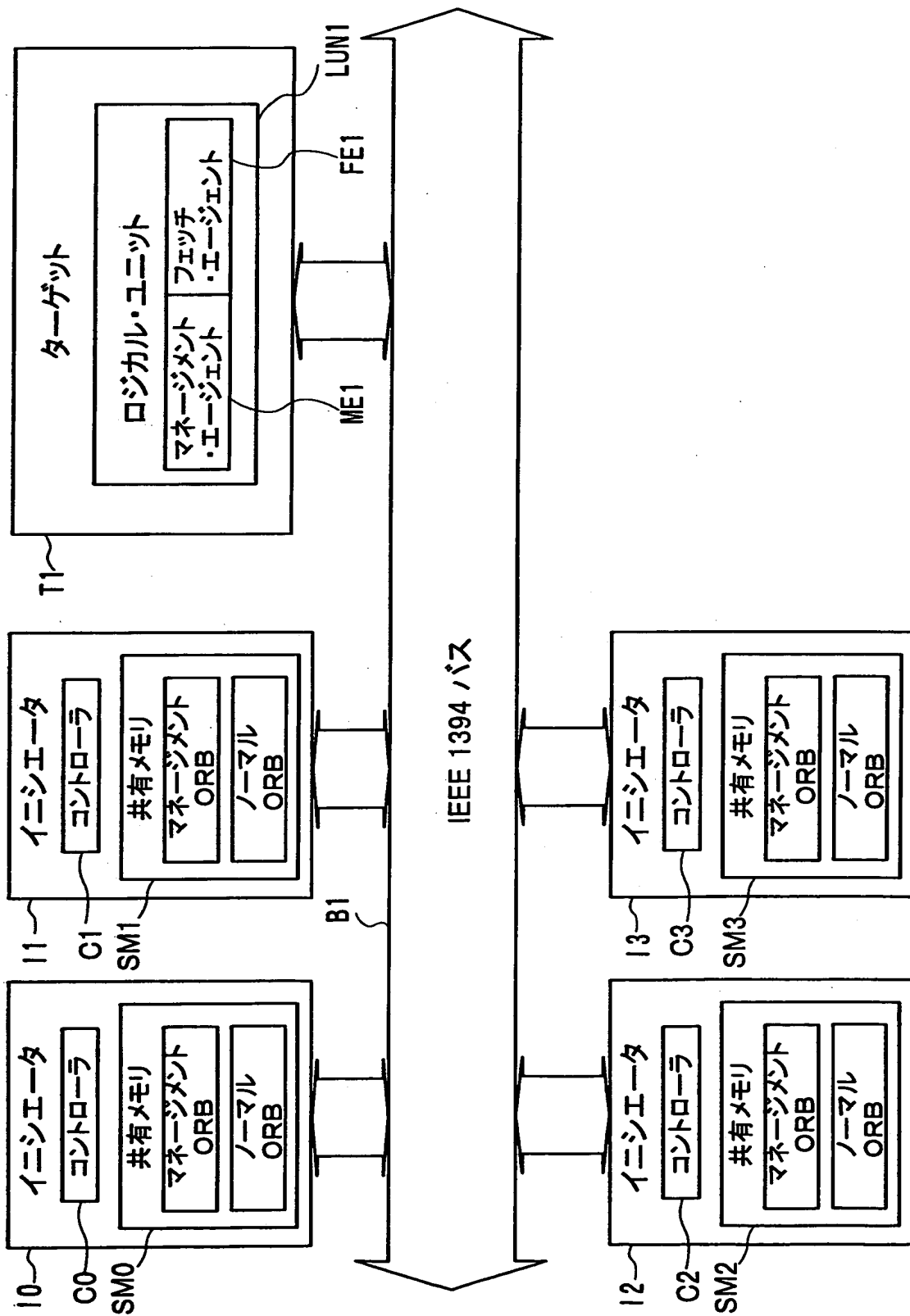
ME1…マネージメント・エージェント

SM0～SM3…共有メモリ

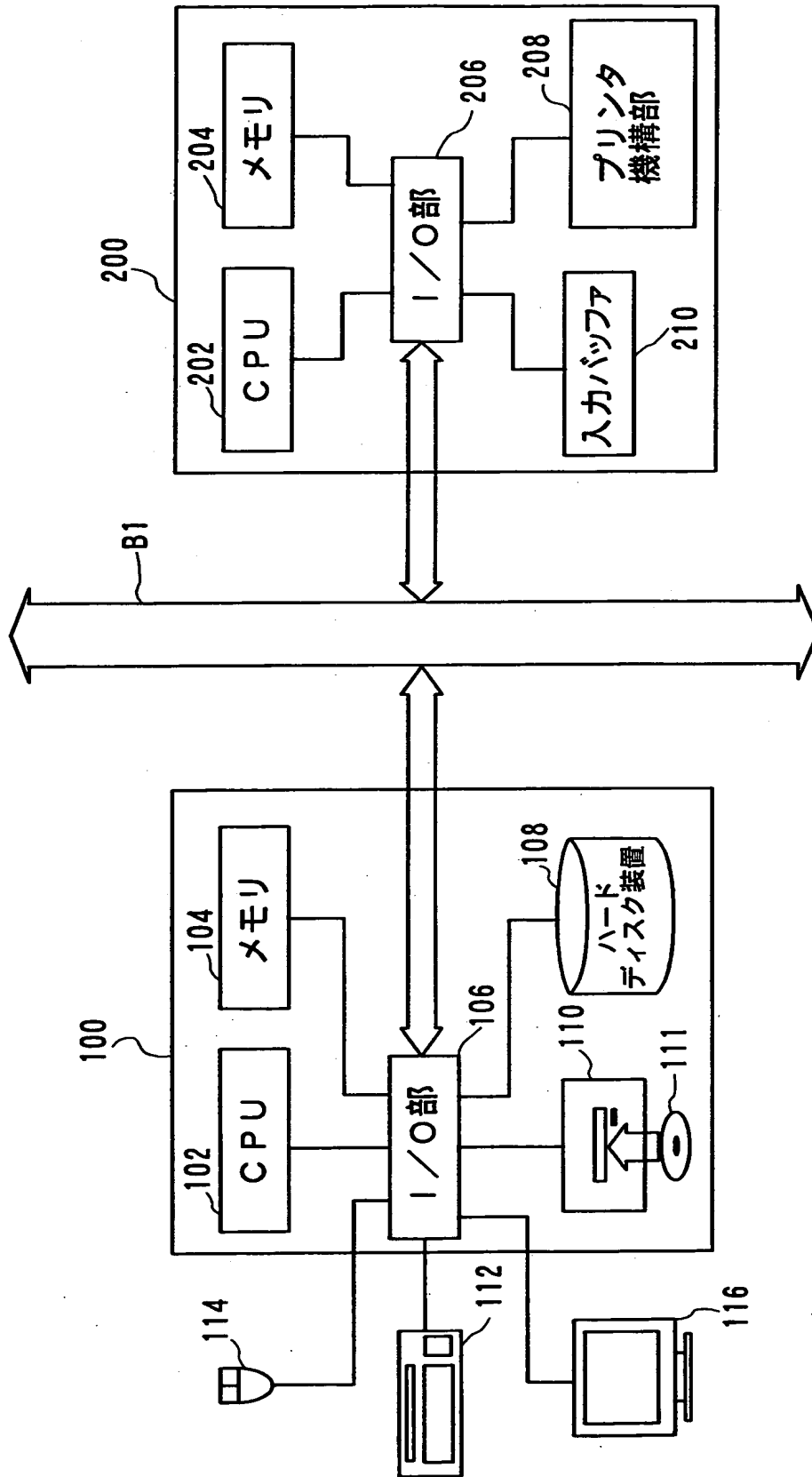
T1…ターゲット

【書類名】 図面

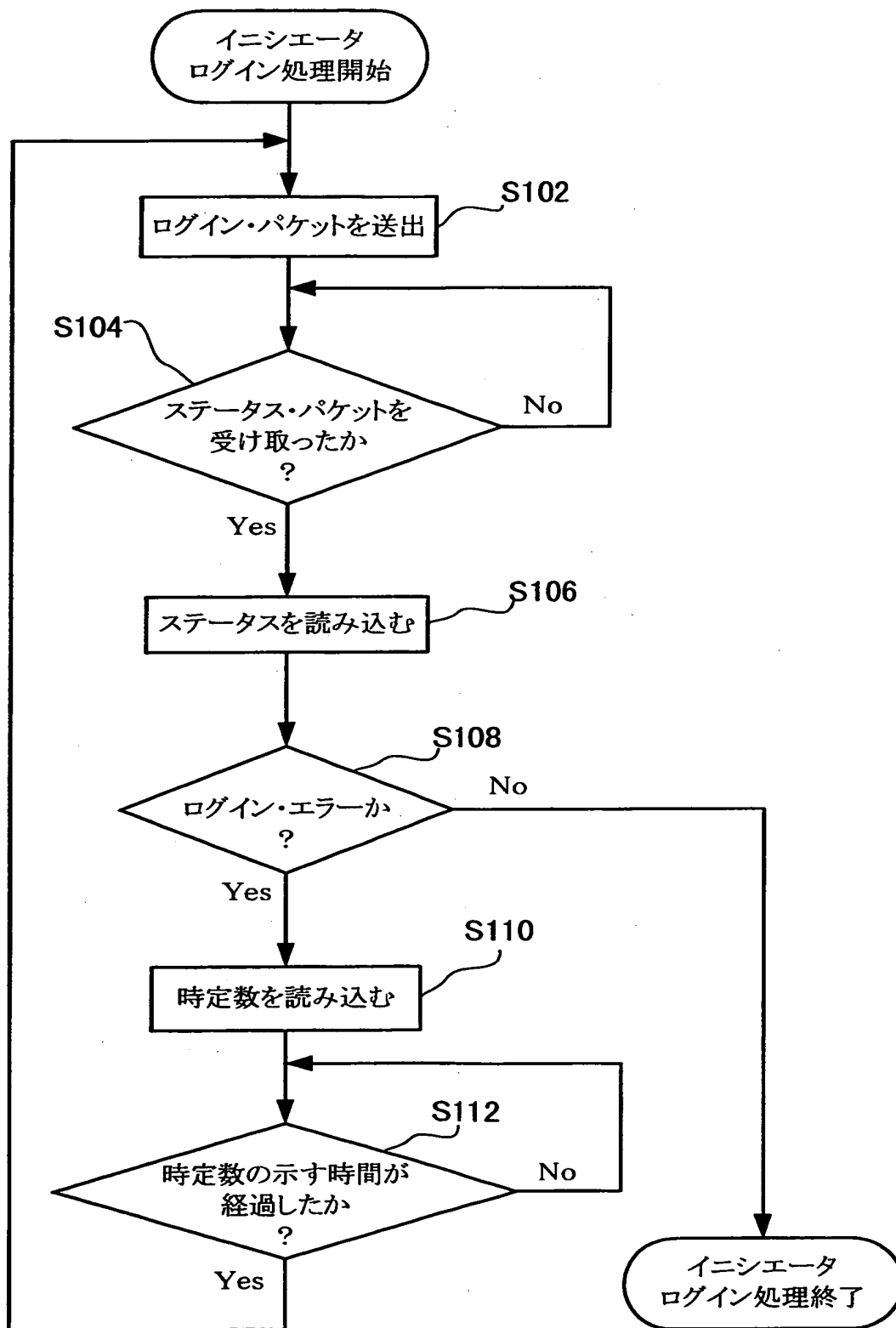
【図 1】



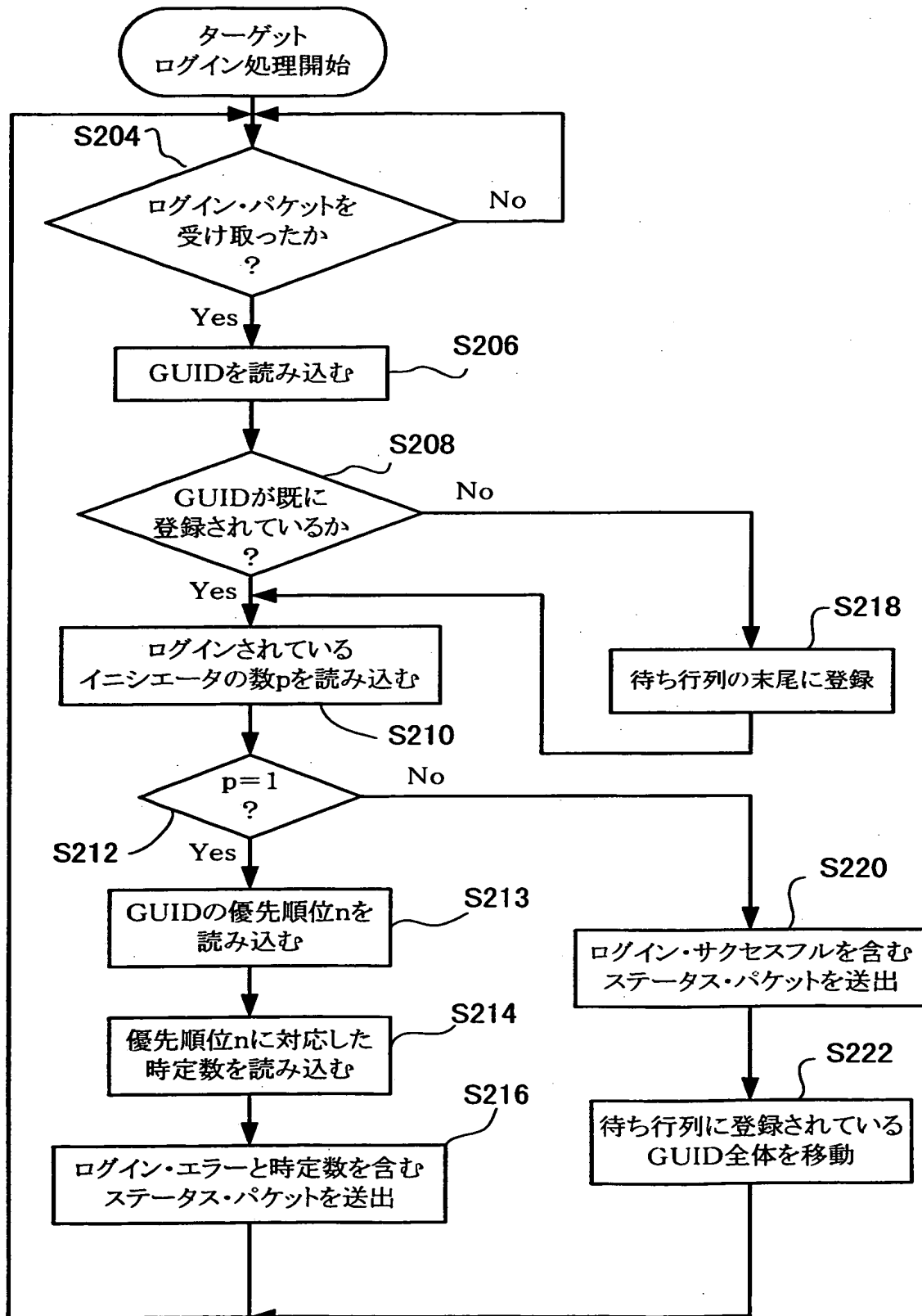
【図 2】



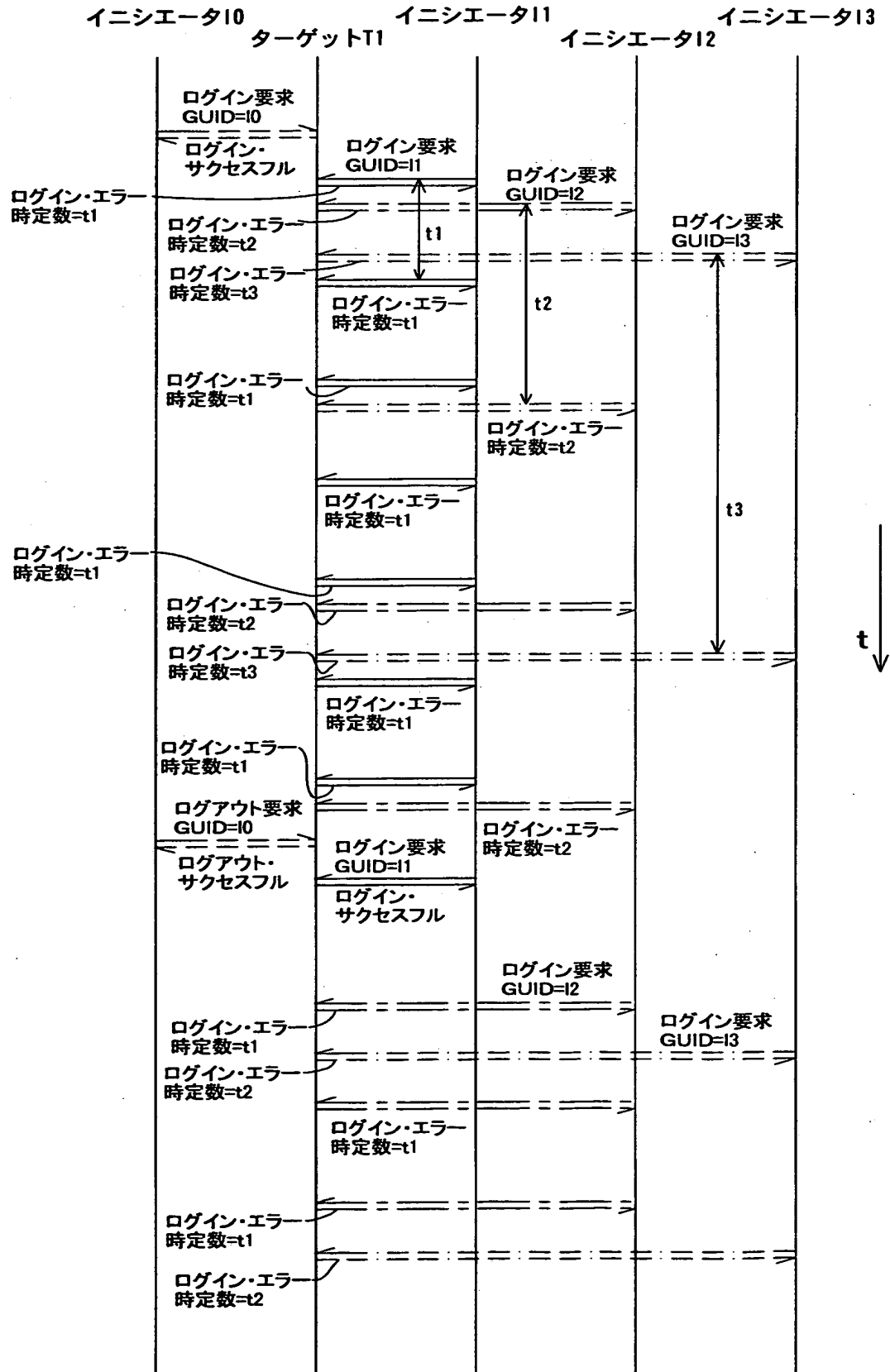
【図 3】



【図 4】



【図 5】



【図 6】

(a)

| n | GUID |
|---|------|
| 0 | |
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |

(b)

| n | GUID |
|---|------|
| 0 | |
| 1 | 10 |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |

(c)

| n | GUID |
|---|------|
| 0 | 10 |
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |

(d)

| n | GUID |
|---|------|
| 0 | 10 |
| 1 | 11 |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |

(e)

| n | GUID |
|---|------|
| 0 | 10 |
| 1 | 11 |
| 2 | 12 |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |

(f)

| n | GUID |
|---|------|
| 0 | 10 |
| 1 | 11 |
| 2 | 12 |
| 3 | 13 |
| 4 | |
| 5 | |
| 6 | |
| 7 | |

(g)

| n | GUID |
|---|------|
| 0 | |
| 1 | 11 |
| 2 | 12 |
| 3 | 13 |
| 4 | |
| 5 | |
| 6 | |
| 7 | |

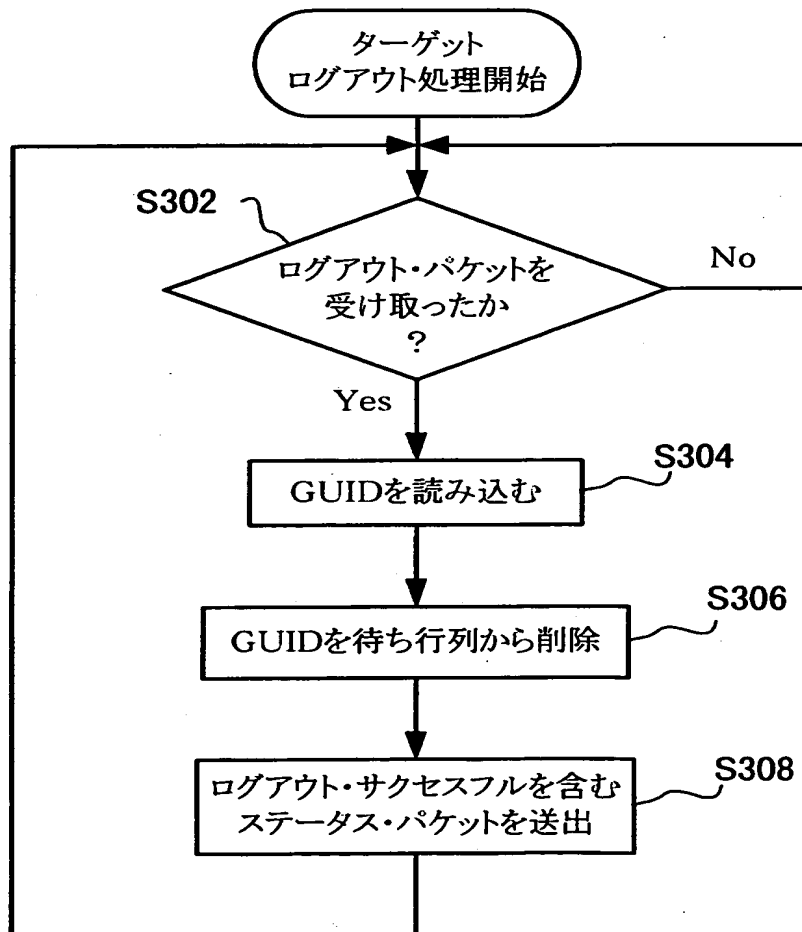
(h)

| n | GUID |
|---|------|
| 0 | 11 |
| 1 | 12 |
| 2 | 13 |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |

【図 7】

| n | 時定数 |
|---|-----------|
| 1 | t1 |
| 2 | t2(=2t1) |
| 3 | t3(=4t1) |
| 4 | t4(=8t1) |
| 5 | t5(=16t1) |
| 6 | t6(=32t1) |
| 7 | t7(=32t1) |

【図 8】



【書類名】 要約書

【要約】

【課題】 ターゲットに対し最先にログイン要求を出したイニシエータが、最先にログインすることを概ね保証することができるようにする。

【解決手段】 ターゲットT1のマネージメント・エージェントME1は、イニシエータからログイン要求を受けた場合、ログインされているイニシエータの数がログイン許容数に対しているか否かを判定する（ステップS210、S212）。達していれば、待ち行列からそのイニシエータのGUIDの優先順位nを読み込み（ステップS213）、時定数テーブルからその優先順位nに対応した時定数を読み込む（ステップS214）。マネージメント・エージェントME1は、「ログイン・エラー」と読み込んだ時定数を含むステータス・パケットを、上記イニシエータに送出する（ステップS216）。そのパケットを受け取ったイニシエータは、そのパケットから時定数を読み込んで、その時定数に応じたタイミングで、再度ログイン要求をターゲットT1に送出する。

【選択図】 図4

出 願 人 履 歴 情 報

識別番号 [000002369]

1. 変更年月日 1990年 8月20日

[変更理由] 新規登録

住 所 東京都新宿区西新宿2丁目4番1号

氏 名 セイコーエプソン株式会社